

## 經濟部智慧財產局專利審定書

中華民國捌拾捌年玖月拾柒日發文

( ) 智專 (二) 04058 字第

號

133459

日期 88.9.20

受文者：滙爾資料公司。

代理人：傅軼群 先生，康偉言 先生

地址：( ) 台北市松山區南京東路三段二四八號七樓

二、申請明案名號數：八六一一七〇四五  
二、發明名稱：從客戶電腦存取主機電腦上之資訊的方法 類：G06F15/163

三、申請人名稱：滙爾資料公司

住址：美國

四、專利代理人姓名：傅軼群 先生，康偉言 先生

住址：台北市松山區南京東路三段二四八號七樓

五、申請日期：中華民國八十六年十一月十四日

六、優先權項目：1996/12/09 08/762,266 美國

七、審查委員姓名：徐演政 委員

八、審定內容：

主文：本案應不予專利。

依據：專利法第二十條第二項

理由：( 接下頁 )。

1. 本案「從客戶電腦存取主機電腦上之資訊的方法」係有關一種控制主機電腦與在遠端客戶電腦間資料傳輸之方法。
2. 本案主要由一顯示控制單元提供該主機電腦與遠端客戶電腦的資訊傳送的媒介，該顯示控制創造數個欄位物件給予鍵盤資料輸入或主機電腦傳送資訊存放，此乃習知技術之使用，我國專利公報公告編號八六一一七〇四五號（如附件所示），其同時可剖析資料以找出各資訊欄位所對應之功能性質，以執行該性質所對應之指令或回應，故本案乃既有知識而為熟悉遠端存取技術者可輕易完成，且未增功效，不具進步性。

據上論結，本案不符法定專利要件，爰依專利法第二十條第二項規定，審定如主文。

局長 陳明邦

如不服本審定，得於文到之次日起三十日內，備具再審查理由

依照分層負責規定  
受權單位主管決行

(11) 公告編號: 273061

(44) 中華民國85年(1996)03月21日

(51) Int. Cl. 5: H04L29/06

12/56

發明

全 13 頁

(54) 名稱: 透過網際網路用於全球資訊網伺服器資料存取之電腦網路

(21) 申請案號: 84109691

(22) 申請日期: 中華民國84年(1995)09月15日

(72) 發明人:

理查·麥可·羅吉

美國

柯納·查理士·拉卡德

美國

(71) 申請人:

萬國商業機器公司

美國

(74) 代理人: 陳長文 先生

1

2

[57] 申請專利範圍:

1. 一種電腦網路, 該種電腦網路包括許多個各支援至少一請求端電腦之伺服器, 該網路包括:

該請求端電腦(130), 用以提出要求;

該伺服器(131), 連接至該請求端做為該請求端之媒介以接收並實行一要求;

許多個資訊存取伺服器(133,133'), 用以在實行要求的過程期間當做該伺服器之次媒介;

該等資訊存取伺服器提供對封裝體物件(圖9及10)之存取, 該等封裝體物件執行在接收到自該伺服器(131)發出的指令時可執行之可程式功能,

該伺服器(131)包括一控制程式媒介(73), 用以接收由請求端電腦(130)提出之使用者對資訊的要求, 並用以將該要求傳輸給一具有封裝體物件(DIS封裝體)之次媒介資訊存取伺服器(133,133'), 該等封裝體物件在該

伺服器(131)要求控制可程式功能時執行;

該等次媒介資訊存取伺服器(133,133')經由網路直接及/或連接至許多個資料庫資源網間連接器(134,134')以便從具有可實行該要求之資料需求的資料之許多個資料庫資源 (DB2、 Oracle、 Sybase、 Redbrick、 Data Warehouse)中的一些擷取資訊;

該等次媒介資訊存取伺服器(133,133')執行一封裝體物件(DIS封裝體)以使包含於該等許多個資料庫資源內實行該要求之一資料需求的任何有關資訊被擷取並由該次媒介封裝體物件處理,

該次媒介在從資料庫擷取並處理該資料以後, 將該經擷取並處理的資料結果儲存於一被產生以返回該伺服器之該控制程式媒介的檔案內, 並返回該被產生之檔案予該伺服器以回應該控

制程式媒介傳輸，

該伺服器之該控制程式媒介在從該次媒介接收到該檔案時使該次媒介之該結果的報告處理成由該請求端要求決定的功能。

2. 根據申請專利範圍第1項之電腦網路，其中

將該次媒介之該結果的報告處理成由該請求端要求決定的功能之處理預設為將該次媒介之處理結果顯示予該請求端供該使用者在一全球資訊網瀏覽器上觀看。

3. 根據申請專利範圍第1項之電腦網路，其中

該控制程式媒介駐存在一可被一請求端電腦使用者用一全球資訊網瀏覽器存取到之伺服器節點位置處，並包括一可由一支援機器環境執行之指令程式以執行包含接收由該請求端電腦提出之要求及藉提供由在次媒介資訊存取伺服器(133,133')處執行一個或更多個封裝體物件而獲得之結果以實行該要求等方法步驟。

4. 根據申請專利範圍第1項之電腦網路，其中

該控制程式媒介駐存在一可被一請求端電腦使用者用一全球資訊網瀏覽器存取到之伺服器節點位置處，且在存取到一網路之伺服器節點時，該伺服器顯示一HTML文件予該請求端電腦使用者；並啟動該控制程式媒介。

5. 根據申請專利範圍第1項之電腦網路，其中

該控制程式媒介駐存在一可被一請求端電腦使用者用一全球資訊網瀏覽器存取到之伺服器節點位置處，且在存取到一網路之伺服器節點時，該伺服器顯示一HTML文件予該請求端電腦使用者；並啟動該控制程式媒介；且一控制程式媒介自HTML文件接收由

使用者輸入之資料並將該使用者輸入之資料在該控制程式媒介啟動時遞交予該控制程式媒介當做其自該HTML文件返回之輸入參數；該等交付予該控制程式媒介之包含來自HTML文件的資料之輸入參數包括要呼叫之命令檔的辨識符號、包含在建構一用來報告要求之結果的HTML文件時所用的HTML陳述之檔案的名稱、及被要求之檔案的型式、和來自HTML文件交付予該控制程式媒介之查詢資料，該查詢資料包含一命令檔程式在其執行前設入之由使用者選擇的值及／或由HTML文件設計者選擇的預設值，所有這些資料均儲存入一緩衝器環境中，且可被該命令檔程式及／或該控制程式媒介存取並被用以設定該命令檔程式的諸變數。

6. 根據申請專利範圍第5項之電腦網路，其中一API組被用以啟動可執行命令檔及經由該等命令檔呼叫之程式。

7. 根據申請專利範圍第5項之電腦網路，其中一呼叫的命令檔之該辨識符號提供往被指認之命令檔的路徑安排。

25. 8. 根據申請專利範圍第1項之電腦網路，其中該控制程式媒介利用一提供執行物件命令檔之API組以提供可程式功能，該等可程式功能包括多重平台上資料庫之佇列及更新功能和對從資料庫擷取之資料執行處理，該等處理包括執行計算、進行格式化、帳戶計費及將結果儲存為一可由控制程式媒介存取之檔案。

9. 根據申請專利範圍第1項之電腦網路，其中

該請求端電腦要求係用一全球資訊網瀏覽器提出，該請求端電腦包括將該全球資訊網瀏覽器與一首頁結合之裝置，包括

40. 一在一首頁節點處之第一控制程式媒

介，其支援連接至該控制程式媒介並支援該首頁且支援一API以存取該第一控制程式媒介節點可用之一資料庫

該控制程式媒介及API讓該全球資訊網瀏覽器之使用者可自該第一控制程式媒介節點可用之該資料庫蒐集資訊並自一內部網路資源蒐集資訊，且回應由一遠端全球資訊網瀏覽器提出之查詢而提供對內部網路資源之存取。

10.根據申請專利範圍第1項之電腦網路，其中包括

自一全球資訊網瀏覽器接收一待實行之要求而當做瀏覽器請求端的媒介之裝置，

一控制程式媒介，用以將分散的諸次媒介組織成支援亦具有一可在網際網路上存取之存取媒介伺服器的全球資訊網伺服器之內部網路上的分散式整合解決伺服器。

11.根據申請專利範圍第10項之電腦網路，其中包括

許多個分散式整合解決伺服器，用來執行經選擇之封裝體物件，該等封裝體物件在從該全球資訊網伺服器控制程式媒介接收到指令時執行諸可程式功能。

12.根據申請專利範圍第11項之電腦網路，其中包括

一連接至許多個資料庫資源之資料庫網間連接器，用來根據自一超文字文件提出之單一要求提供來自位於在地理上分散的不同型式資料庫處之多重資料基礎的被要求資訊。

13.根據申請專利範圍第12項之電腦網路，其中包括

命令物件，該等命令物件用來在報告至全球資訊網瀏覽器或其他位置以前執行計算、格式化、及其他服務，其報告是依一被要求之結果報告由一組

結果報告中選出之被選擇格式，其包含有連同帳戶記錄之顯示報告、傳真報告、印表機報告、往顧客設備的報告、及往電視視訊訂戶之報告。

5. 14.根據申請專利範圍第1項之電腦網路，其中包括

在該請求端電腦處用以提出要求之全球資訊網瀏覽器，

藉著連接或用一統一資源定位器定址，將該全球資訊網瀏覽器與一在該伺服器處之首頁關聯的裝置，

一位在網際網路上某處之該伺服器節點處的控制程式媒介，藉著連接或用一統一資源定位器定址而支援連接之該控制程式媒介並支援該首頁，

15. 該控制程式媒介伺服器用一內部網路內提供之功能經由一網路連接有擁有者專屬功能且可受內部網路邊界處之防火牆保護，

20. 該控制程式媒介連接至一做為命令檔伺服器之資訊存取伺服器，且該命令檔伺服器連接至一資料庫網間連接器以自連接至該資料庫網間連接器且位於不同資料庫伺服器上的資料庫蒐集資訊，該命令檔伺服器支援許多個命令檔物件，該等命令檔物件被程式安排依該全球資訊網瀏覽器使用者要求執行全球資訊網瀏覽器服務支援功能以在內部網路內存取資訊並做為該控制程式媒介的一個次媒介而透過網際網路蒐集位在其他處所之資訊。

15.根據申請專利範圍第14項之電腦網路，

其中藉著送出一全球資訊網瀏覽器之要求，一使用者不但可存取內部網路內之資訊，而且還能擴展到內部網路之外而透過網際網路蒐集位在他處之資訊。

16.根據申請專利範圍第14項之電腦網路，

40.

其中在包括一內部網路與網際網路之網路上有許多個資料庫網間連接器，且在一執行於一命令檔伺服器內之命令檔的指令下，一資料庫網間連接器藉一網路間路由通信協定經由網路連接至另一資料庫網間連接器。

17. 根據申請專利範圍第14項之電腦網路

其中在包括一內部網路與網際網路之網路上有許多個資料庫網間連接器，且在一執行於一命令檔伺服器內之命令檔的指令下，一資料庫網間連接器藉著一靠UAL啟動資料庫網間連接器連接之網路間路由通信協定經由網路連接至另一資料庫網間連接器。

18. 根據申請專利範圍第16項之電腦網路

其中一由全球資訊網瀏覽器提出之要求被經由一內部網路散佈到網際網路，藉此不但可在內部網路存取資料，還可經由網際網路自位於內部網路之外受一命令檔伺服器支援的資料庫蒐集資料。

19. 根據申請專利範圍第16項之電腦網路

其中一由全球資訊網瀏覽器提出之要求被經由一內部網路散佈到網際網路，藉此不但可在內部網路存取資料，還可經由網際網路自位於內部網路之外受一命令檔伺服器支援且用公眾存取權或在處理透過該命令檔伺服器提供之可變存取授權資料後獲得之存取權連接至該命令檔伺服器的資料庫蒐集資料。

20. 根據申請專利範圍第1項之電腦網路

一在該請求端電腦處之全球資訊網瀏覽器，用來提出要求，藉著連接或用一統一資源定位器定址，將該全球資訊網瀏覽器與一在該伺

服器處之首頁關聯的裝置，

一位在網際網路上某處之該伺服器處的第一控制程式媒介，藉著連接或用一統一資源定位器定址而支援連接之該控制程式媒介並支援該首頁，

5.

一位在網際網路上某處之第二控制程式媒介節點，藉著連接或用一統一資源定位器定址而支援一第二控制程式媒介，

10.

該第二控制程式媒介節點用一內部網路內提供之功能經由一網路連接有擁有者專屬功能且可受內部網路邊界處之防火牆保護，

15.

該第一控制程式媒介連接至位於網際網路上某處之該第二控制程式媒介節點，其支援該第二控制程式媒介且連接至並支援一命令檔伺服器，該命令檔伺服器連接至一資料庫網間連接器以便自連接至該資料庫網間連接器並位於不同的資料庫伺服器上之資料庫蒐集資訊，該命令檔伺服器支援許多個命令檔物件，該等命令檔物件被程式化安排以根據該全球資訊網瀏覽器使用者的要求執行全球資訊網瀏覽器服務支援功能以在內部網路內存取資訊並做為該控制程式媒介之次媒介而經由網際網路蒐集位於他處之資訊。

20.

25.

21. 根據申請專利範圍第20項之電腦網路

30.

其中該第一控制程式媒介駐存在一支援該全球資訊網瀏覽器之第一全球資訊網伺服器上，且該第二控制程式媒介駐存在一第二全球資訊網伺服器上，該第二全球資訊網伺服器經由其本身的網路連接至一關聯的命令檔伺服器以執行由該全球資訊網瀏覽器要求的任務並在通過多重網路之後傳送至該全球資訊網瀏覽器。

35.

22. 根據申請專利範圍第1項之電腦網路

40.

，包括

用以將該全球資訊網瀏覽器與一首頁關聯之裝置，包括

一第一控制程式媒介節點，其支援一連接的控制程式媒介且支援該首頁並支援一API以存取該第一控制程式媒介節點可用之資料庫，

該控制程式媒介及API使該全球資訊網瀏覽器使用者可自該第一控制程式媒介節點可用之該資料庫蒐集資訊並自一內部網路資源蒐集資訊且依據一由遠端全球資訊網瀏覽器提出之查詢要求而提供對該內部網路資源之存取。

### 23. 根據申請專利範圍第22項之電腦網路，

其中該遠端全球資訊網瀏覽器亦連接至一位於網際網路上的第二控制程式媒介節點，該第二控制程式媒介節點支援一支援一API之第二控制程式媒介以經由該第二控制程式媒介存取該第一控制程式媒介節點可用的一資料庫，

該第二控制程式媒介及API使該全球資訊網瀏覽器使用者可經由該第二控制程式媒介節點自該第一控制程式媒介節點可用的一資料庫蒐集資訊並自一內部網路資源蒐集資訊，且依據一由遠端全球資訊網瀏覽器藉著一連接或用一統一資源定位器定址至該第二控制程式媒介節點而提出之查詢要求提供對該內部網路資源之存取及從連接至該第二控制程式媒介節點之內部網路上可用的資源存取資料。

### 24. 根據申請專利範圍第22項之電腦網路，其中

該第二控制程式媒介節點用一內部網路內提供之功能經由一網路連接有擁有者專屬功能且可受內部網路邊界處之防火牆保護，

該位於網際網路上某處之第二控制程

式媒介節點藉連接或用一統一資源定位器定址而支援該第二控制程式媒介，

該第一控制程式媒介連接至該位於網際網路上某處之第二控制程式媒介節點，其支援該第二控制程式媒介且連接至並支援一命令檔伺服器，該命令檔伺服器連接至一資料庫網間連接器以便自連接至該資料庫網間連接器並位於不同的資料庫伺服器上之資料庫蒐集資訊，該命令檔伺服器支援許多個命令檔物件，該等命令檔物件被程式化安排以根據該全球資訊網瀏覽器使用者的要求執行全球資訊網瀏覽器服務支援功能以在內部網路內存取資訊並做為該控制程式媒介之次媒介而經由網際網路蒐集位於他處之資訊。

### 25. 根據申請專利範圍第22項之電腦網路，

其中該全球資訊網瀏覽器位在一全球資訊網伺服器處，該全球資訊網伺服器提供該控制程式媒介節點，且瀏覽器要求若經授權跨越該內部網路存取則存取一提供該第二命令檔媒介節點之該內部網路上一全球資訊網伺服器內的一命令檔媒介，該命令檔媒介接著利用由一DIS伺服器提供之DIS封裝體而作用如一命令檔伺服器。

### 26. 根據申請專利範圍第1項之電腦網路，

其中該控制程式媒介為了控制接收及實行全球資訊網瀏覽器之要求而包括，

一為了實行連接至一網路之全球資訊網瀏覽器請求端的要求之服務媒介；包括一以可見方式實踐一可由支援機器環境執行的指令程式之控制程式媒介，該指令程式用來執行接收由該全球資訊網瀏覽器請求端提出之要求並藉提供一結果以實行該要求的方法步驟，該方法步驟包括：

將一HTML文件顯示至該全球資訊網瀏覽器；

啟動一控制程式媒介；

自HTML文件接收使用者輸入之資料並在該控制程式媒介啟動時將該使用者輸入之資料交給該控制程式媒介做為該控制程式媒介自該HTML文件返回之輸入參數；

使用一API組啟動可執行命令檔及可經由一關聯之命令檔物件存取之程式。

27. 根據申請專利範圍第26項之電腦網路，其中

該等輸入參數包含來自HTML文件交付予該控制程式媒介之資料。

28. 根據申請專利範圍第26項之電腦網路，其中該自HTML文件接收使用者輸入之資料並在該控制程式媒介啟動時將該使用者輸入之資料交給該控制程式媒介做為該控制程式媒介自該HTML文件返回之輸入參數的步驟包括要呼叫之命令檔的辨識符號、包含在建構一用來報告要求之結果的HTML文件時所用的HTML陳述之檔案的名稱、及被要求之檔案的型式、和來自HTML文件交付予該控制程式媒介之查詢資料。

29. 根據申請專利範圍第26項之電腦網路，其中該將來自HTML文件之資料交付予該控制程式媒介的步驟包括交付資料，該資料包含一命令檔程式在其執行前設入之由使用者選擇的值及／或由HTML文件設計者選擇的預設值，所有這些資料均儲存入一緩衝器環境中，且可被該命令檔程式及／或該控制程式媒介存取並被用以設定該命令檔程式的諸變數。

30. 根據申請專利範圍第26項之電腦網路，其中該方法步驟尚包括：

在被該控制程式媒介處理期間提供設

定給發生的API呼叫。

31. 根據申請專利範圍第26項之電腦網路，其中該方法步驟尚包括：

在被該控制程式媒介處理期間根據要求指令提供一結果，該結果之預設是返回全球資訊網瀏覽器首頁。

32. 根據申請專利範圍第1項之電腦網路，其中該控制程式媒介以可見方式實踐一可由支援機器環境執行的指令程式，該指令程式用來執行接收由該全球資訊網瀏覽器請求端提出之要求並藉提供一結果以實行該要求的方法步驟，該方法步驟包括：

將一HTML文件顯示至該全球資訊網瀏覽器；

啟動一控制程式媒介；

自HTML文件接收使用者輸入之資料並在該控制程式媒介啟動時將該使用者輸入之資料交給該控制程式媒介做為該控制程式媒介自該HTML文件返回之輸入參數；並將該等輸入參數儲存入一緩衝器；

用控制程式媒介自一儲存體擷取命令檔及關聯於各命令檔之變數名稱並將可用的命令檔名稱及關聯於各命令檔之變數名稱載入關聯於控制程式的記憶體內；

用控制程式媒介啟動介於該控制程式媒介與一API之間的連接；

一旦一命令檔被載入該控制程式媒介之記憶體內，該控制程式媒介即自其記憶體擷取在該控制程式媒介被啟動時交付予該控制程式媒介之關聯於一命令檔名稱的命令檔變數名稱；

接著，用該控制程式媒介產生一儲存於控制程式記憶體內之資料陣列，該資料陣列包含在初始資料擷取步驟期間交付予控制程式媒介之命令檔變數名稱及其值，以便將命令檔變數名稱與用該環境變數步驟交付予控制程式

媒介之資料匹配；

接著在準備一報告時，該程式命令媒介產生一獨特檔名，該獨特檔名可包括由在一以前步驟(步驟112)中儲存之HTML文件的變數產生之資料，該獨特檔名被交付予命令檔當做為由該命令檔產生之報告命名使用之命令檔變數，結果是命令檔在此程序期間將產生具有該獨特檔名之檔案；

接著在預期命令檔執行時，被命令檔使用之諸變數值從控制程式記憶體內之資料陣列獲得，該等變數值包括命令檔變數名稱及其值，該等資料是在自HTML文件提供交付予該控制程式媒介之查詢資料參數的步驟中交付予控制程式媒介的，該等參數包括一命令檔程式在其執行前設入之由使用者選擇的值及／或由HTML文件設計者選擇的預設值，使用API以設定命令檔變數值來進行此項工作；

接著使用一API以查詢並決定被要求工作佇列的大小，並比較該大小是否超過預定臨限數目個被要求之工作，且若是，則進入一等待狀態迴圈以等待並再反覆比較該大小是否仍超過該預定臨限數目個被要求之工作，直到佇列大小縮減至低於該臨限位準，然後在該時點使用該API送出一命令檔供執行；

在該API送出該命令檔供一次媒介執行以後，該控制程式媒介進入一狀態等待該命令檔被該次媒介執行；

且其中在控制程式媒介讀取被指認之檔案的步驟期間，其根據全球資訊網瀏覽器之要求動態地產生一結果報告之新HTML陳述。

- 33.根據申請專利範圍第32項之電腦網路，其中該方法步驟尚包括：用該新HTML陳述擷取自該檔案擷取之資訊以使其可被供以一命令檔報告之結果

- 34.根據申請專利範圍第33項之電腦網路，其中該方法步驟尚包括：用該控制程式媒介藉從被儲存之變數獲取資訊並辨識輸出參數而測試要產生的報告種類，且分支往可應用於要產生的報告種類之序列。

- 35.根據申請專利範圍第33項之電腦網路，其中該命令檔媒介是許多個由一支援機器環境執行的可程式指令，用來執行由一全球資訊網瀏覽器請求端提出之要求所要求之方法步驟並藉提供一結果以實行該要求，該方法步驟包括：

10. 在該命令檔媒介處接收一來自控制程式媒介準備一報告之委託命令及關聯於一報告而要交付予命令檔做為用以對要由該命令檔產生之報告命名所使用之命令檔變數，其結果是命令檔在該命令檔媒介執行期間將產生該具有獨特檔名之檔案，

根據該命令檔指定之分散式資料環境內一特定順序流程由該命令檔媒介處理一系列的經連結之物件，包括執行在該命令檔媒介內指定的包含資料擷取及處理之功能，及

在該處理程序中產生一具有該獨特檔名之報名檔案，並在該命令檔媒介執行期間將該處理結果儲存入該具有該獨特檔名之報名檔案內。

20. 36.根據申請專利範圍第33項之電腦網路，其具有一種方法，該方法包括下列方法步驟：

在執行一由該控制程式媒介委託之要求期間，在接收到授權資料之後，由該命令檔次媒介啟動網路間路由分配功能。

35. 37.根據申請專利範圍第33項之電腦網路，其具有一種方法，該方法包括下列方法步驟：



在產生一具有該獨特檔名之報告檔案之前，接收由該控制程式媒介儲存之由一HTML文件的變數產生之資料。

38.根據申請專利範圍第33項之電腦網路，其具有一種方法，該方法包括下列方法步驟：

啟動網路間路由分配功能以查詢資料庫其位在網路上何處。

39.根據申請專利範圍第33項之電腦網路，其具有一種方法，該方法包括下列方法步驟：

藉著由一介於諸系統間之連結聯絡網際網路上的一目的地並根據可用的通信協定將從一實體單元擷取之資料路由分配至另一實體單元而啟動網路間路由分配功能以查詢資料庫其位在網路上何處。

40.根據申請專利範圍第1項之電腦網路，

其中該伺服器是一支援全球資訊網瀏覽器之全球資訊網伺服器，

且該伺服器包括

自一全球資訊網瀏覽器接收一待實行之要求而當做瀏覽器請求端之媒介的裝置，

一控制程式媒介，用以組織分散的諸次媒介成為支援全球資訊網伺服器之內部網路上的分散式整合解決伺服器，該全球資訊網伺服器亦有一可在網際網路上存取之存取媒介伺服器。

41.根據申請專利範圍第40項之電腦網路，尚包括

許多個用以執行被選擇之封裝體物件的分散式整合解決伺服器，該等對裝體物件在從該全球資訊網伺服器控制程式媒介接收到指令時執行可程式功能。

42.根據申請專利範圍第41項之電腦網路，尚包括

一連接至許多個資料庫資源之資料庫

網間連接器，用以依據來自HTML文件提出之單一要求從位於在地理上分散之不同種類資料庫處之多重資料基礎供應被要求的資訊。

5. 43.根據申請專利範圍第41項之電腦網路，尚包括

命令物件，用以在向全球資訊網瀏覽器或其他位置依被選擇格式報告以前執行計算、格式化、及其他服務，該被選擇格式係由一組結果報告中選擇之被要求的結果報告格式，該組結果報告包括具備帳戶記錄之顯示報告、傳真報告、印表機報告、往客戶設備之報告、及一往電視視訊用戶之報告。

15.

圖示簡單說明：

圖1以簡圖顯示較佳具體實例的概觀，並特別顯示一典型的分散式電腦系統，該系統有包括網際網路之內部及外部網路以將請求端連接至全球資訊網伺服器及本發明所位在之系統內的其他伺服器。

20.

圖2顯示當請求端由一全球資訊網瀏覽器連接至其伺服器(可能是一網際網路伺服器)之後顯示在請求端上的查詢螢幕畫面(首頁)。

25.

圖3是次一螢幕畫面，該畫面顯示如何根據使用者期望提出一要求，根據本發明以一所示的輸入畫面提出一要求。

30.

圖4是在被要求之服務由根據本發明之電腦系統網路依據一DIS封裝體規格格式化提供之後返回予請求端之結果螢幕畫面樣本。

35.

圖5是次一畫面，該畫面說明如何根據使用者期望提出一要求，根據本發明藉自一選單選擇並透過使用影像對應提出一要求。

40.

圖6是在被要求之服務由根據本發明之電腦系統網路提供以後返回予請求端之圖形結果畫面的一種範列。

圖7是一流程圖，顯示一網路伺服器與諸如IBM之資料解譯系統(DIS)的決策支援系統工具之間的資料流程，並顯示全球資訊網請求端至全球資訊網伺服器之連接及一要求至執行一DIS封裝體之連接，和全球資訊網伺服器內自ODAS至一具有異質連接到多重資料庫之分散式DIS LAN的連接。

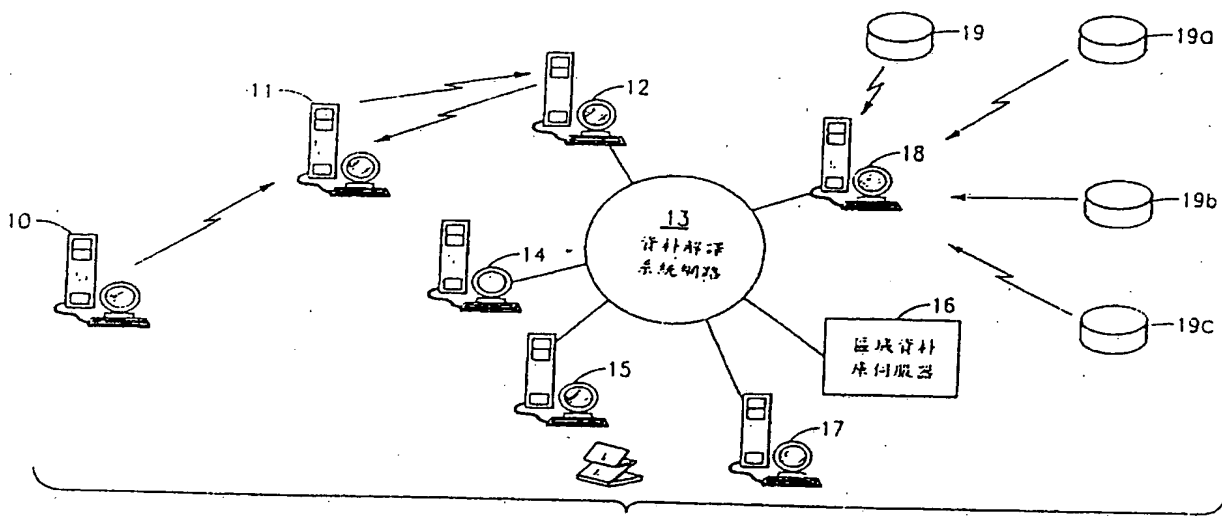
圖8顯示網路伺服器控制程式諸功能

之流程圖。

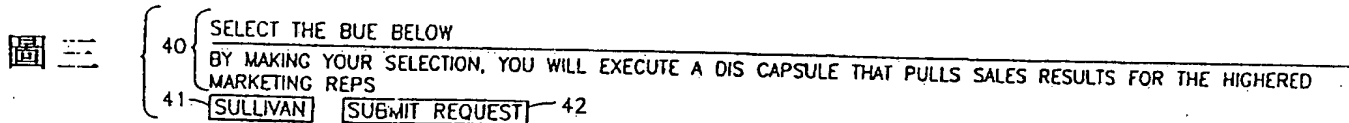
圖9藉範例顯示產生文字報告檔案之DIS封裝體。

圖10藉範例顯示產生圖形報告檔案之DIS封裝體。

圖11顯示網路系統的一種替代性架構，其可被用以容許透過首頁存取資料倉內可用的資訊，其中對首頁或資料庫之存取可以或不可以受限於防火牆。



圖一



WELCOME TO THE HOMEPAGE OF...



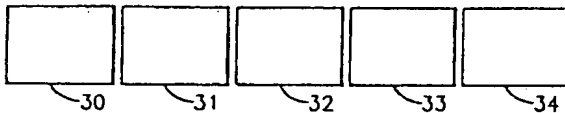
THE CURRENT DATE IS FRI APR 28 11:05:38 1995

THIS PAGE IS UNDER DEVELOPMENT! (C)COPYRIGHT IBM CORPORATION, 1994, 1995. ALL RIGHTS RESERVED.

"WE'RE COMING TO LIVE FROM THE ON-RAMP TO THE INFORMATION SUPER HIGHWAY." THIS IS THE IBM U.S. HIGHER EDUCATION HOME PAGE ON THE WORLD WIDE WEB.

BY ACCESSING THIS PAGE YOU WILL HAVE THE UNIQUE CAPABILITY OF OBTAINING INTERACTIVE INFORMATION RELATING TO OUR BUSINESS. BY MERELY CLICKING ON THE HIGHLIGHTED TEXT AND FOLLOWING THE ROAD-SIGNS YOU'LL BE WELL ON YOUR WAY TO INFOCENTRIC CITY!!

MAKE YOUR SELECTION BELOW



圖二

FILE NAME IS P81484. ←43

CATCH IT ON THE NET!

44. YOU JUST RAN A DATA INTERPRETATION SYSTEM CAPSULE!  
THE OUTPUT IS THE TEXT TOOL TRANSFERRED AS A TEXT.  
COPYRIGHT IBM 1995

CUSTOMERS WHO MIGHT WANT TO KNOW ABOUT THE  
DIS WWW GATEWAY---THIS DATA IS FROM THE IBM MARKET PLANNING DATA SYSTEM---  
A DB2 MVS DATABASE

CUSTOMER NAME	SW DESCRIPTION	INST	CITY STATE
AC NIELSEN CO	DIS 1.3.6 DIS ENTERPRISE SE	9501	CHERRY HILL NJ
AC NIELSEN CO	DIS 1.3.6 DIS ENTERPRISE SE	9501	GREEN BAY WI
AC NIELSEN CO	DIS 1.3.6 DIS ENTERPRISE SE	9501	MINNEAPOLIS MN
AC NIELSEN CO	DIS 1.3.6 DIS ENTERPRISE SE	9501	WILTON CT
ADVANTIS	DIS 1.3.6 DIS ENTERPRISE SE	9501	SCHAUMBURG IL
ALTA BATES MEDI	DIS 1.3.6 DIS ENTERPRISE SE	9410	BERKELEY CA
ALTA BATES MEDI	DIS 1.3.6 DIS ENTERPRISE SE	9503	BERKELEY CA
AMERICAN PRESID	DIS 1.3.6 DIS ENTERPRISE SE	9501	OAKLAND CA
ANHEUSER BUSCH	DIS 1.3.6 DIS ENTERPRISE SE	9501	ST LOUIS MO
ANHEUSER BUSCH	DIS 2.0 OPENDIS ACCESS SERV	9501	ST LOUIS MO
50. ASHLAND OIL INC	DIS 1.3.6 DIS ENTERPRISE SE	9502	LEXINGTON KY
ASHLAND OIL INC	DIS 2.0 OPENDIS ACCESS SERV	9502	LEXINGTON KY
BELLSOUTH CELLU	DIS 1.3.6 DIS ENTERPRISE SE	9501	FT LAUDERDALE FL
BELLSOUTH CELLU	DIS 2.0 OPENDIS ACCESS SERV	9501	FT LAUDERDALE FL
BELLSOUTH COMMU	DIS 1.3.6 DIS ENTERPRISE SE	9501	ATLANTA GA
BELLSOUTH COMMU	DIS 1.3.6 DIS ENTERPRISE SE	9501	BIRMINGHAM AL
BELLSOUTH TELEC	DIS 1.3.6 DIS ENTERPRISE SE	9501	ATLANTA GA
BRIQ TECHNOLOGY	DIS 1.3.6 DIS ENTERPRISE SE	9501	MOUNTAIN VIEW CA
BRISTOL MYERS S	DIS 1.3.6 DIS ENTERPRISE SE	9410	PLAINSBORO NJ
BRISTOL MYERS S	DIS 2.0 OPENDIS ACCESS SERV	9502	NEW YORK NY
BROADWAY DEPT S	DIS 1.3.6 DIS ENTERPRISE SE	9410	LOS ANGELES CA
BROOKLYN UNION	DIS 1.3.6 DIS ENTERPRISE SE	9410	BROOKLYN NY
CHESEBROUGH PON	DIS 1.3.6 DIS ENTERPRISE SE	9501	GREENWICH CT
COLGATE-PALMOLI	DIS 1.3.6 DIS ENTERPRISE SE	9410	IRVINE CA
COLGATE-PALMOLI	DIS 1.3.6 DIS ENTERPRISE SE	9410	MORRISTOWN NJ
COLGATE-PALMOLI	DIS 1.3.6 DIS ENTERPRISE SE	9410	NEW YORK NY

圖四

SELECT THE BUE BELOW

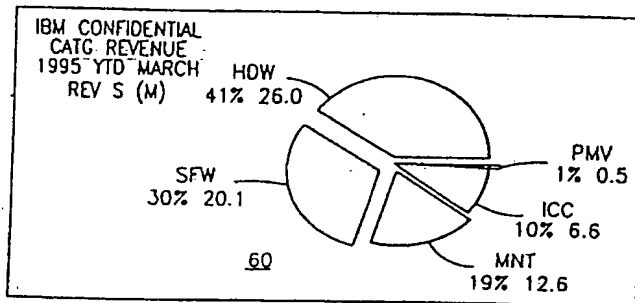
40A BY MAKING YOUR SELECTION, YOU WILL EXECUTE A DIS CAPSULE THAT PULLS SALES RESULTS FOR YTD CATALOG REVENUE.

41   42

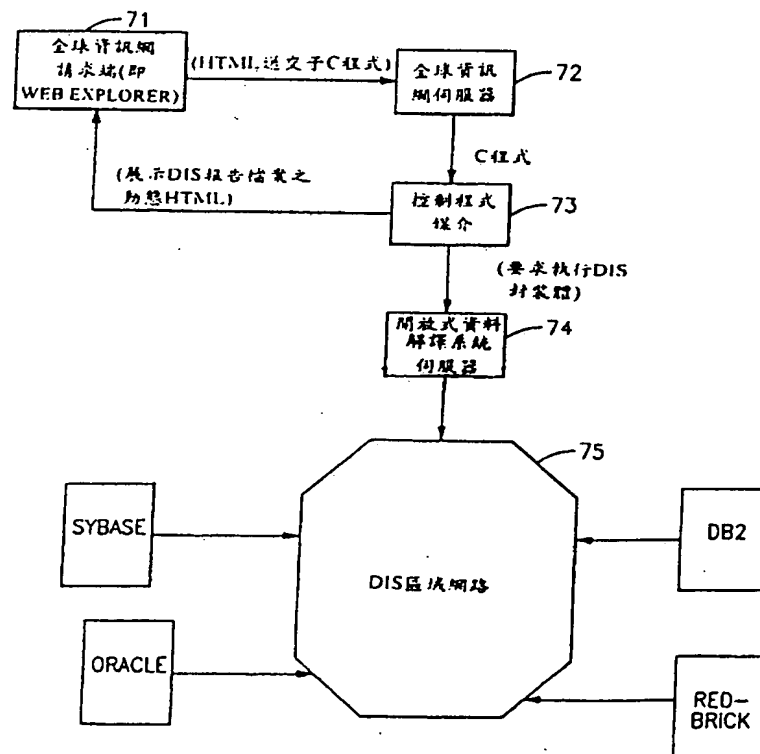
圖五

FILE NAME IS .P555119. ←59

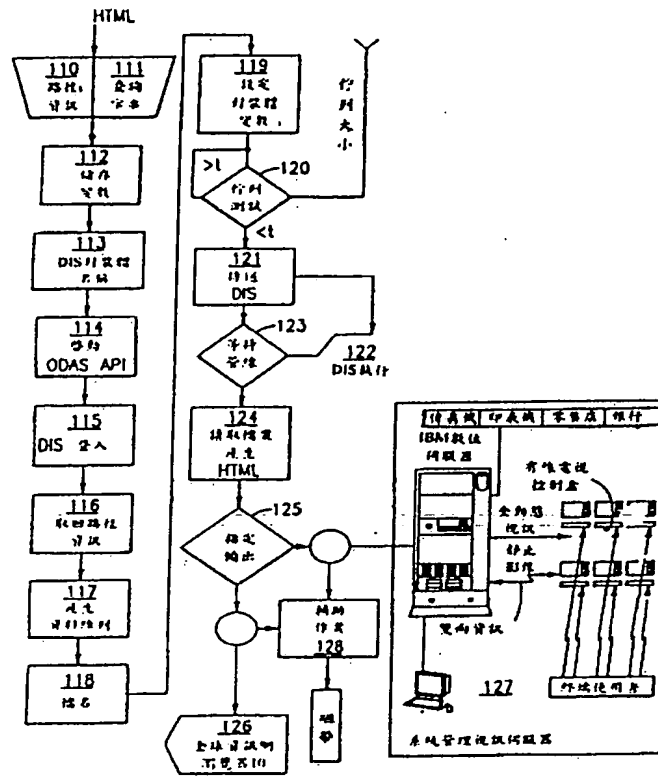
61 { CATCH IT ON THE NET!  
YOU JUST RAN A DATA INTERPRETATION SYSTEM CAPSULE! THE OUTPUT IS THE PLOT TOOL TRANSFERRED AS A BITMAP.  
COPYRIGHT IBM 1995



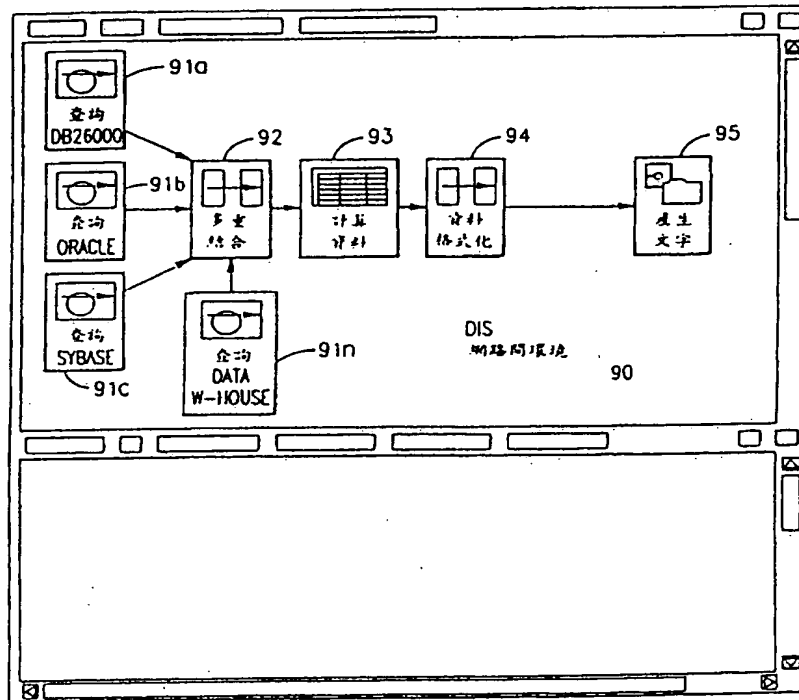
圖六



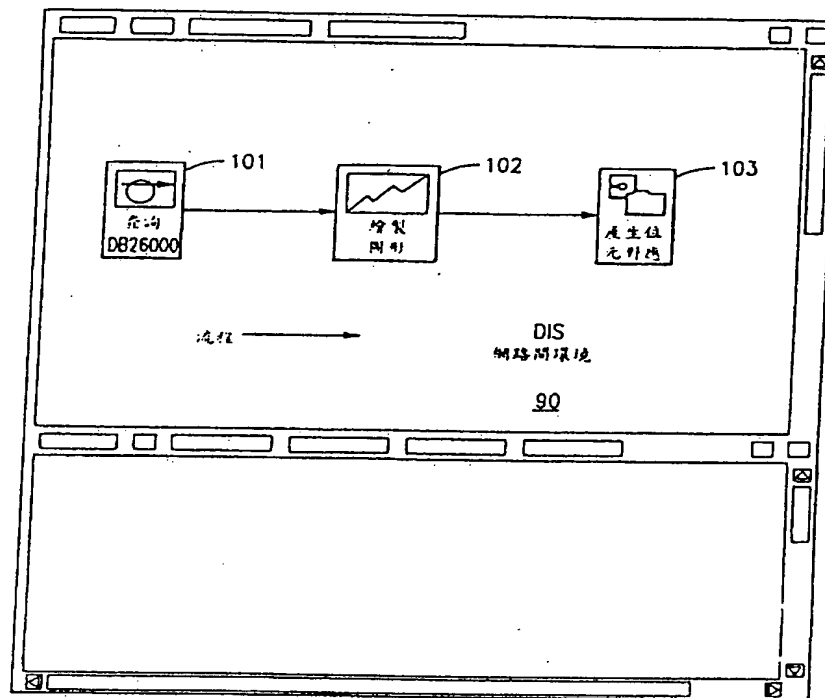
圖七



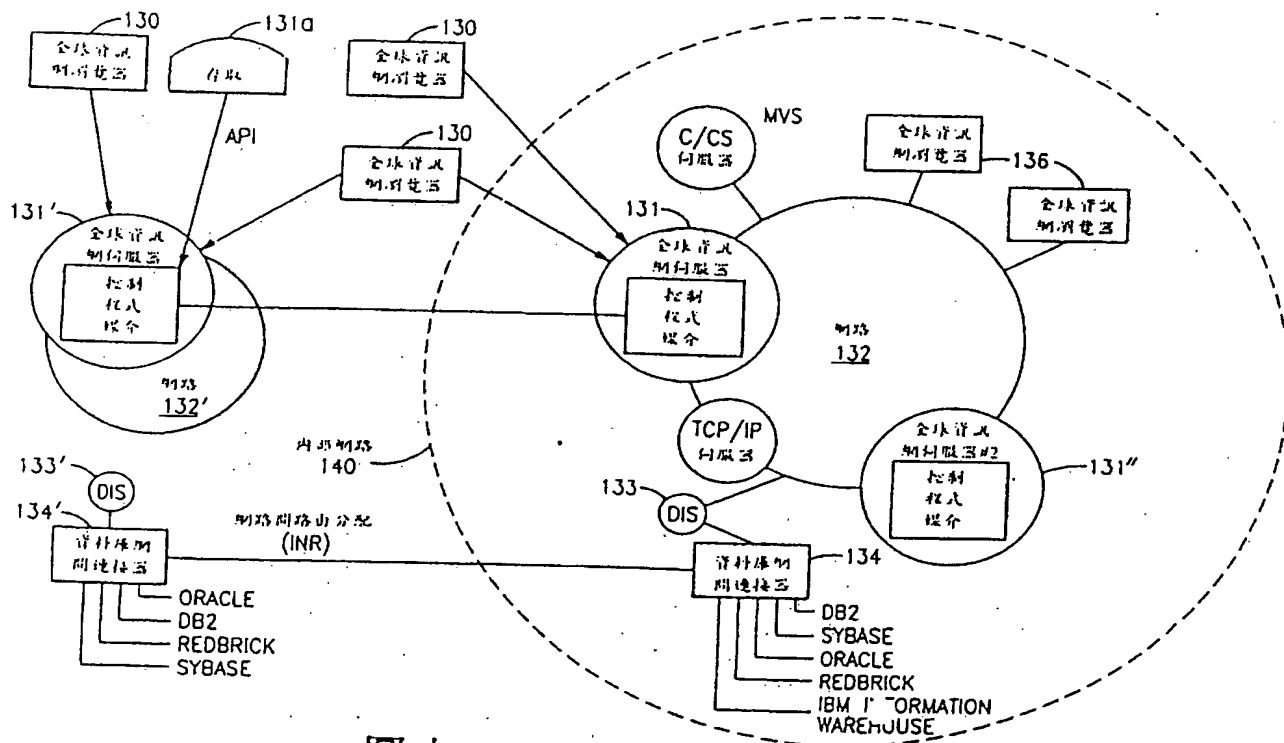
圖八



圖九



圖十



圖十一

## ABSTRACT

A World Wide Web browser makes requests to web servers on a network which receive and fulfill requests as an agent of the browser client, organizing distributed sub-agents as distributed integration solution (DIS) servers on an intranet network supporting the web server which also has an access agent servers accessible over the Internet. DIS servers execute selected capsule objects which perform programmable functions upon a received command from a web server control program agent for retrieving, from a database gateway coupled to a plurality of database resources upon a single request made from a Hypertext document, requested information from multiple data.bases located at different types of databases geographically dispersed, performing calculations, formatting, and other services prior to reporting to the web browser or to other locations, in a selected format, as in a display, fax, printer, and to customer installations or to TV video subscribers, with account tracking.

---

## TITLE: Computer network for WWW Server Data Access over Internet

---

---

### Copyright Authorization

A portion of the disclosure of this patent document contains material which is subject to copyright protection. The owner, International Business Machines Corporation, has no objection to the facsimile reproduction by any one of the patent disclosure, as it appears in the Patent and Trademark Office patent files or records of any country, but otherwise reserves all rights whatsoever.

---

### FIELD OF THE INVENTION

This invention is related to computers and computer systems and particularly to a method and system for use of the World Wide Web and other sources of information and for utilization of existing equipment advantageously for web server data access over networks and the Internet.

---

### Related Applications

This application entitled "Computer Network for WWW Server Data Access over Internet", is related to other United States of America Patent applications filed concurrently herewith, and specifically to the applications entitled "A Service Agent for Fulfilling requests of a Web Browser", USSN 08/474,576, filed June 7, 1995; and "A Sub-Agent Service Agent for Fulfilling Requests of a Web Browser", USSN 08/474,575, filed June 7, 1995; and "A Web Browser System", USSN 08/479,481, filed June 7, 1995; and "A Method for Fulfilling Requests of a Web Browser" USSN 08/474,577, filed June 7, 1995; and "A Method for Distributed Task Fulfillment of Web Browser Requests", USSN 08/474,572, filed June 7, 1995.

These applications have a common assignee: International Business Machines Corporation, Armonk, New York.

---

### GLOSSARY OF TERMS

While dictionary meanings are also implied by certain terms used here, the following glossary of some terms may be useful.

#### World Wide Web (WWW)

The Internet's application that lets people seeking information on the Internet switch from server to server and database to database by clicking on highlighted words or phrases of interest. An Internet WWW server supports clients and provides information.

#### Home page

A multi-media table of contents that guides a web user to stored information about an organization on the Internet.



- Gopher** A menu-based search scheme, which as developed at the University of Minnesota, lets a user reach a destination on the Internet by selecting items from a series of text menus.
- Access Agent** A logical component that provides support for different access protocols and data streams -- Frame Relay, HDLC (High Data Link Control) CBO (Continuous bit Operations, ATM (Asynchronous Transfer Mode), or TCP/IP.
- Application Processing Agent**  
A data processing agent running in a server data processing system which performs tasks based on received requests from a client in a distributed environment. In our preferred embodiment, our application processing agent for database retrieval is our DIS server, a data interpretation system server and database gateway which is coupled to our web server HTTPD via a network. In our preferred embodiment an application processing agent employs executable object programs as command file objects, which in the preferred embodiment are capsule objects.
- Client** A client is a computer serviced by the server which provides commands to the server.
- Data Interpretation System (DIS).**  
IBM's object oriented decision support tool.
- Capsule** A DIS capsule is a program created by a DIS programmer and executed in the DIS environment. A DIS capsule is a preferred example of a capsule object. A capsule object is a specialized form of a command file (which is a list of commands to be executed, as in an EXEC or \*.BAT batch file. The capsule object is created with an object environment, as is supplied by IBM's DIS. Other object environments are IBM's SOM and DSOM, and Microsoft's COM environment.
- Internet** The connection system that links computers worldwide in a web.
- Server** A machine which supports one or more clients and is part of the web. Any computer that performs a task at the command of another computer is a server.
- Slip or PPP connection.**  
Serial-line Internet protocol and point-to-point protocol, respectively, for providing a full access connection for a computer to the Internet.
- TCP/IP** Transmission control protocol/Internet protocol. A packet switching scheme the Internet uses to chop, route, and reconstruct the data it handles, from e-mail to video.

**InterNetwork Routing (INR)**

The link between systems which routes data from one physical unit to another according to the applicable protocol. The protocol will employ a URL address for Internet locations.

**URL**

Universal resource locator, a Web document version of an e-mail address. URLs are very cumbersome if they belong to documents buried deep within others. They can be accessed with a Hyperlink.

**Web browser**

An program running on a computer that acts as an Internet tour guide, complete with pictorial desktops, directories and search tools used when a user "surfs" the Internet. In this application the Web browser is a client service which communicates with the World Wide Web.

**HTTPD**

An IBM OS/2 Web Server or other server having Hypertext Markup Language and Common Gateway Interface. In our preferred embodiment, the HTTPD incorporates our control program agent and is supported by an access agent which provides the hardware connections to machines on the intranet and access to the Internet, such as TCP/IP couplings.

**HTTP Hypertext transfer protocol**

Hypertext transfer protocol. At the beginning of a URL "http:" indicates the file contains hyperlinks.

**Hyperlink**

A network address embedded in a word, phrase, icon or picture that is activated when you select the highlighted tidbit. Information about that item is currently retrieved to the client supporting a Web browser.

**HyperText Markup Language (HTML)**

HTML is the language used by Web servers to create and connect documents that are viewed by Web clients. HTML uses Hypertext documents. Other uses of Hypertext documents are described in U.S. Patents 5,204,947, granted April 20, 1993 to Bernstein et al.; 5,297,249, granted March 22, 1994 to Bernstein et al.; 5,355,472, granted October 11, 1994 to Lewis; all of which are assigned to International Business Machines Corporation, and which are referenced herein.

---

**BACKGROUND OF THE INVENTION**

The Internet is not a single network, it has no owner or controller, but is an unruly network of networks, a confederation of many different nets, public and private, big and small, that have agreed to connect to one another. An intranet is a network which is restricted and while it may follow the Internet protocol, none or only part of the network available from outside a "firewall" surrounding the intranet is part of the agreed connection to the Internet. The composite network represented by these net-

works relies on no single transmission medium, bi-directional communication can occur via satellite links, fiber-optic trunk lines, phone lines, cable TV wires and local radio links. When your client computer logs onto the Internet at a university, a corporate office or from home, everything looks local, but the access to the network does cost time and line charges.

Until recently, "cruising or surfing" the Internet was a disorienting, even infuriating experience, something like trying to navigate without charts. The World Wide Web, a sub-network of the Internet, introduced about two years ago, made it easier by letting people jump from one server to another simply by selecting a highlighted word, picture or icon (a program object representation) about which they want more information -- a maneuver known as a "hyperlink". In order to explore the WWW today, the user loads a special navigation program, called a "Web browser" onto his computer. While there are several versions of Web browsers, IBM's example is the new WebExplorer which offers users of IBM's OS/2 Warp system software a consistent, easy to use desktop of pictorial icons and pull down menus. As part of a group of integrated applications available from IBM for OS/2 Warp called the IBM Internet Connection, lets users log onto the Internet.

To this point the World Wide Web (Web) provided by Internet has been used in industry predominately as a means of communication, advertisement, and placement of orders. As background for our invention there now exists a number of Internet browsers. Common examples are NetScape, Mosaic and IBM's Web Explorer. Browsers allow a user of a client to access servers located throughout the world for information which is stored therein and provided to the client by the server by sending files or data packs to the requesting client from the server's resources. An example of such a request might be something called GSQL (get SQL) which was a NCSA language and CGI server program developed to getting textual results for a client caller. Developed by Jason Ng at the University of Illinois, this document provided a way to map SQL forms against a database, and return the textual results to the client caller. This system is unlike the present invention, and presents difficulties which are overcome by our described system.

These servers act as a kind of Application Processing Agent, or (as they may be referred to) an "intelligent agent", by receiving a function request from a client in response to which the server which performs tasks, the function, based on received requests from a client in a distributed environment. This function shipping concept in a distributed environment was first illustrated by CICS as a result of the invention described in U.S. Patent 4,274,139 to Hodgkinson et al. This kind of function, illustrated by CICS and its improvements, has been widely used in what is now known as transaction processing. However, servers today, while performing many functions, do not permit the functions which we have developed to be performed as we will describe.

Now, "surfing" the Internet with the WWW is still a time consuming affair, and the information received is not generally useful in the form presented. Even with 14,400 baud connection to the Internet much line time is tied up in just keeping going an

access to the Internet, and the users don't generally know where to go. Furthermore the coupling of resources available on a company's intranet and those available on the Internet has not been resolved. There is a need to reduce gateways, make better use of existing equipment, and allow greater and more effective usage of information which is resident in many different databases on many different servers, not only within a homogeneous network but also via the Internet and heterogeneous network systems.

The problems with creating access to the world via the Internet and still to allow internal access to databases has been enormous. However, the need for a system which can be used across machines and operating systems and differing gateways is strongly felt by users of the Internet today. Anyone who has spent hours at a WWW browser doing simple task knows how difficult it still is to navigate thorough arcane rules without knowing where to go and even if you know what you are doing spending hours doing routine tasks. Many needs exist. As one important instance, until now we know of no way to access data on multiple databases of different types using a single user request from a client. This and other difficulties are solved by our invention.

---

## SUMMARY OF THE INVENTION

In accordance with our invention needless user intervention is eliminate or greatly reduced with a Web server supports an HTTPD which is provided with the capabilities of our control program agent which organizes sub-agents supporting command file objects or capsules to perform tasks in support of a Web browser's request for service as programmable functions receiving parameters as input and providing as their output handled by the control program agent task completed results for reporting in accordance with the Web browser request in the form and to the location determined by a request and handling these request without needless user intervention.

In accordance with our invention, we have created a way to allow Web users to request information that is created by a data interpretation system (DIS) and then presented by a web server to the user of the web. Our solution provides a way of requesting and processing and presenting information on the Web. In the process, data is retrieved from multiple sources which may be located remotely and accessed via an intranet routing and via the Web Internet and processed by our decision support capsules. Now companies and universities, and other users that want to access data located on different databases, want that data processed and formatted, and presented in a form the user desires, such as a graphical format. Our solution permits users to access information from various sources and obtain information at a desired location as a result of a single request which is responded to by an organization of facilities and command file sub-agent decision support capsule objects by our command program agent. Users of the information can be internal to a company, or external. The result can be furnished to a user at a location which is internal or external to the company, and as specified at a specified location with a form and format desired. This allows a report to be managed by the web support services we

provide, and in a form consistent with the request, but without requiring a consistent interface solution.

In order to create a way for Web users to request information generation we provide a web server with a control program agent which is linked to a decision support tool of a data interpretation system server, the application processing agent, and then have that server retrieve, process, and format information which is presented to the user on the Web by the Web server. In our preferred embodiment, we have provided a link between a Hypertext Markup Language (HTML) document using a common gateway interface, and open data interpretation system server (ODAS). As a result, Web clients can request DIS reports to be generated, specify the parameters to be used in generating the reports, and then view the report results on a Web home page. The DIS capsule can generate graphical information, such as colored pie charts, line graphs, bar graphs, and other forms of generated information. Since the Web server is capable of presenting the results in desired formats, the full capabilities of a DIS report are utilized.

Our invention provides a method and system for allowing a user of a client to access and assemble information structured and reported to the user in accordance with his desires, selecting information for disparate servers which are located within a network can be an intranet or internal network, such as a LAN or WAN not normally accessible to the Internet, or coupled to the Internet. In accordance with our invention one can access data on multiple databases of different types using a single user request from a client. We also allow the facility for providing specialized specific requests to be created for routine use, as well as the facility to formulate generalized or specialized ad hoc requests. In addition, we provide besides query and update capability, the ability to perform calculations with respect to any retrieved data, to format the information in text or in graphics, and the facility of presenting the results to the client for display or other use.

The improvements which we have made achieve a means for accepting Web client requests for information, obtaining data from one or more databases which may be located on multiple platforms at different physical locations on an Internet or on the Internet, processing that data into meaningful information, and presenting that information to the Web client in a text or graphics display as a location specified by the request.

Our invention of providing a web server with a control program agent allows organization of decision support functions to be executed by application processing agent servers located throughout the Internet to gather and supply information not presently available with any existing resources without the need of endless intervention on the part of a requesting user of the WWW; further enabling an ordinary user to take advantage of expertise which is provided by programmable sub-agents developed by those with particular expertise in a given area as well as enabling use of standard routines commonly needed.

These improvements are accomplished by providing for Web clients to request information from an application processing agent in which the application processing agent server performs tasks based on received requests from a client in a distributed environment by a web server supported by an access agent link and control program agent which in turn causes a decision support function to be executed by the application processing agent server. This is performed within the distributed environment by the application processing agent server which forms part of a network coupled to and under control of the control program agent. According to our invention the decision support function is provided by a data interpretation system which functions as part of the application processing agent and the decision support function is programmable and generated by a data interpretation system, DIS or other decision support element performing similar functions, and provided in a form accessible to our control program agent which presents the output generated to be presented to the user on the Web who made the initial request. We have provided, in a preferred embodiment, a link between IBM's Hypertext Markup Language (HTML), the Common Gateway Interface (CGI), and the Open DIS Access Server (ODAS), all of which may be used on machines which are commercially available from IBM. In order to write additional functions which develop our invention, the reader is referred to the Medaphor Data Interpretation System publication "Developing Applications with OpenDIS Access Service, Version 2.0, available from IBM, First Edition (September 1994) Part Number 315-0002-01 which is incorporated herein by reference.

Our improvements relating to our control program agent is in accordance with our preferred embodiment is normally installed on an IBM HTTPD which is an IBM OS/2 Web Server or other server having Hypertext Markup Language and Common Gateway Interface. In our preferred embodiment, the HTTPD incorporates our control program agent and is supported by an access agent which provides the hardware connections to machines on the intranet and access to the Internet, such as TCP/IP couplings. The hardware for the Web server is thus a workstation, such as IBM's PS/2 model 80 with OS/2. However, the HTTPD can be installed in PCs and upwardly also in machines which range across IBM's line of computers from powerful personal computers to mainframe systems which support MVS. IBM's operating system which enables multiple kinds of operating systems, including "UNIX" to co-exist on a single platform. As a result of our invention Web clients can request DIS reports to be generated by the application processing agent specifying the parameters to be used in generating the reports, and then as a result of the request receive a result which is presented, as a visual display or otherwise, on a Web page for use by the requesting user. Our machine implementation allows a user having DIS access to generate graphical information such as colored pie charts, line graphs, bar graphs, etc. Since Web browsers such as IBM's Web Explorer are capable of displaying these formats, all the functions which can be created by a DIS capsule can be utilized by a user of our invention.

According to our improved method, an Internet World Wide Web user connects to a Web server through the use of a Web browser. In accordance with our preferred embodiment, we use HTML as the language used by Web servers to create and connect documents that are viewed by Web clients. HTML is an example of a

hypertext language having the facility of clicking on a highlighted word, string of words, or image in order to move to another HTML document or invoke a program on the server. An example of a Web client would be a machine used by a person using IBM's Web Explorer product. In using our invention a user may click on the hypertext in a document to reference a function which will be provided by an application processing agent server. The user is able to connect to another document that may be on another Web server. HTML commands are used to reference other documents. HTML is used to reference programs available on a server, and pass parameters to those programs. The application processing agent server executes a program when it is referred to by a Web client via a control program agent resident, preferably, in a Web server.

The Web client selects the information that they wish to view by using the HTML created page, the Web server takes the client request and passes it to a C program implementation of our control program agent. Web servers, such as HTTPD for OS/2, with our control program agent are able to provide access to executable programs through the use of the Common Gateway Interface (CGI). When a program is referenced by the HTML, any parameters are passed to the program and it is executed. In our preferred embodiment we have used CGI to invoke programs that we have developed that will interface with the DIS product. CGI is an example of a software gateway from a Web server to programs outside the Web server application.

The control program agent that is called in this instance by the Web server through the CGI interface, passes the Web client request along to a data interpretation system DIS via a Open Dis Access Server ( ODAS ). ODAS is a feature of a data interpretation system DIS that allows programs to initiate DIS functions, such as invoking DIS capsules. Our control program agents interface with DIS through ODAS to submit DIS capsules for execution. DIS capsules are basically programs that DIS application programmers create with the DIS programming language. In accordance with our invention, we have written capsules which are executed as a DIS capsule on a server to gather data from one or more databases, process that data, and create a report in one of many formats, which we will describe by way of example. After the DIS capsule completes executing, in accordance with our preferred embodiment, the results that are generated during execution of a capsule are stored in a file on the application processing server.

After DIS creates a file that contains the formatted report results, our control program agents program dynamically creates HTML tags to present the formatted report back to the Web client on the Internet. Our control program agents using the CGI interface can create HTML commands dynamically. In this way a program can present information on a Web browser for the Web client.

After the DIS capsule has created the file containing the report request results, the control program creates HTML statements dynamically that display the report results to the Web browser.

Alternative means of presenting the data are shown by alternative routing. The user requesting the report may wish to have the report results sent to another location in addition to or instead of displaying the report results to the Web browser. This information is provided during the request phase. As a result of the alternative report request, and according to the parameters indicated therein, the report results can be sent by the control program via electronic mail, i.e. TCP/IP Sendmail facility and Lotus Notes, to one or more locations on the Internet. The report results can be sent as a file and as a note. The request can request a voice response, which can be routed to a voice response unit. Thus, with a call to a translator, the text can be converted to voice, and even translated along the way. The report results can also be sent to a fax machine, or to a computer that has the capability of receiving fax data.

We use these report concepts to present report files created by DIS capsules on the Web client display.

These and other improvements are set forth in the following detailed description. For a better understanding of the invention with advantages and features, refer to the description and to the drawings.

---

## BRIEF DESCRIPTION OF THE DRAWINGS.

FIGURE 1 shows schematically an overview of the preferred embodiment and particularly shows a typical distributed computer system which has internal and external networks including the Internet to connect clients to World Wide Web servers and other servers within the system in which our invention is situated.

FIGURE 2 shows a inquiry screen (home page) which is displayed on a client after the client is coupled to its server (which may be an Internet server) by a Web browser.

FIGURE 3 is a next screen which illustrates how a request is made according to a users desires, making a request in accordance with our invention with an input screen shown.

FIGURE 4 is a sample result screen which is returned to the client after the requested service is provided by the computer system network in accordance with our invention formatted according to the specifications of a DIS capsule.

FIGURE 5 is a next screen which illustrates how a request is made according to a users desires, making a request in accordance with our invention by selection from a menu and through the use of image mapping.

FIGURE 6 is an example of a graphical result screen which is returned to the client after the requested service is provided by the computer system network in accordance with our invention.



FIGURE 7 illustrates a flowchart showing data flow between a web server and decision support system tool such as IBM's Data Interpretation System (DIS), and shows the coupling of a Web client to a Web server and the coupling of a request to execute a DIS capsule and the coupling within the Web server from ODAS to a distributed DIS LAN with heterogeneous connections to multiple databases.

FIGURE 8 illustrates as a flow chart the functions of the control program for the web server.

FIGURE 9 illustrates by way of example a DIS capsule that creates a text report file.

FIGURE 10 illustrates by way of example a DIS capsule that creates a graphical report file.

FIGURE 11 illustrates an alternative configuration of the network system as it may be employed for permitting access to information available through homepages and in data warehouses where access to the homepage or database may or may not be restricted by a firewall.

(Note: For convenience of illustration, in the formal drawings FIGURES may be separated in parts and as a convention we place the top of the FIGURE as the first sheet, with subsequent sheets proceeding down and across when viewing the FIGURE, in the event that multiple sheets are used.)

Our detailed description explains the preferred embodiments of our invention, together with advantages and features, by way of example with reference to the following drawings.

---

## DETAILED DESCRIPTION OF THE INVENTION

Figure 1 illustrates a information delivery solution of a typical combination of resources including clients and servers which may be personal computers or workstations as clients, and workstations to mainframe servers as servers. The various elements are coupled to one another by various networks, including LANs, WANs, and other networks, which may be internal SNA networks or other like internal networks, and also providing access to the Internet, which couples the system to the world via Internet.

## The Preferred Embodiment

Turning now to our invention in greater detail, it will be seen from FIGURE 1 that our preferred embodiment provides a Web browser 10, which is coupled to a Web server 11. Our Internet WWW browser is an intelligent computer system, such as an IBM PS/2, or other computer, an IBM ThinkPad, an RS/6000 works as well and connections are made to the network via OS/2 WARP Connect, an IBM product. The Internet Web browser in the intelligent computer system which performs the Web browser function has IBM Web Explorer, or NetScape or Mosaic installed thereon. This computer system 10 is bi-directionally coupled with the OS/2 WARP Connect facility over a line or via a wireless system to our preferred computer system which we call our Web server. This system is a PS/2 or RS/6000 or other similar system which includes our control program agent 73, which will be discussed below. Web server 11, in our preferred embodiment is coupled again bi-directionally via a line or wireless coupling to a computer system, such as a PS/2 or RS/6000 or other server which supports and performs the server function of ODAS server 12, which is coupled to the distributed DIS network, here shown as LAN 13. ODAS 12 may be located on the same server as the Web server 11 or be located at a separate service machine, such as an IBM Digital Server. The Web server is logically coupled to our application processing agent server via a network. We call our application processing agent server a DIS File server 14 because it comprises a data interpretation system which supports the decision support functions we provide which is today most inexpensively provided by an IBM computer system which supports OS/2. In our preferred embodiment, the intranet network is a LAN. Thus the components of the DIS LAN 13 comprise a DIS File Server 14, a general purpose workstation 15 which can be used for capsule development, a local database server 16, a Capsule Server 17 for storing a plurality of DIS capsules ready for user, a Database Gateway Server 18 which performs the gateway functions to access databases which are linked to it, these databases include geographically distributed databases which can be located, for instance, in Chicago, New York, Dallas, Los Angeles, and each of which can be a different supported database, such as DB2 database 19, ORACLE database 20, Sybase database 21, Redbrick database 22. In our preferred embodiment all servers are coupled with a conventional LAN or WAN connection, with a preferred IBM token ring shown. Reference should also be had to our alternative preferred embodiment discussed below with respect to FIGURE 11.

Thus, in connection with the preferred embodiment of FIGURE 1 as well as with respect to FIGURE 11 it would be appreciated from the schematic overview illustrated by FIGURE 1 and FIGURE 11 that our invention may be employed in a distributed computer system environment which has internal or intranet networks represented in our preferred embodiment by the DIS Network 13 and external networks including the Internet to connect clients to World Wide Web servers and other servers within the system in which our invention is situated. Our invention makes use of the entire network. The Web browser 10 can make a request to the Web Server 11 for a report. The Web server 11 with the facilities we provide causes the application processing agent which includes our DIS server 14 and its supporting communication server, the database gateway server 18, to act as an agent to gather data from one or more of the

multiple databases, including the local database 16, DB2 database 19, ORACLE database 20, Sybase database 21, Redbrick database 22. Further details with respect to the use of our invention for database retrieval of information from multiple databases are provided as to the actions of the application processing agent functions of the database server(s) 18 with reference to FIGURE 7.

Thus, returning to our simplified and preferred embodiment, FIGURE 2 shows a inquiry screen (home page) 29 in the form which is displayed on a client after the client is coupled to its server (which may be an Internet Web server 11) by a Web browser 10. The entire screen contains information and a plurality of objects. Once the home page is displayed, with appropriate descriptive guidance as illustrated by the FIGURE 2, the user can interact, for example, by clicking on image objects 30, 31, 32, 33, 34. As a example should the user want to make a special request in accordance with our invention, he could click on image 30. This would take the user to the next screen, illustrated by FIGURE 3. Alternatively the user could select by clicking on image 31 another menu screen, illustrated by FIGURE 5. At this point also, a specialized format could be selected by double clicking first on a format select image illustrated by image objects representing access to menu screens 32, 33, 34, one or more of which a gopher.

The use of selection of icon image object is a function provided by HTML and programmers knowing this language can readily create variants to the images and functions we have illustrated. Thus incorporated within the drawings are to be understood to be the variants that can thus be created using our examples, as well as extensions and combinations thereof.

When the user selected image 30 by clicking on the image 30, FIGURE 3 appears. FIGURE 3 is the next screen which illustrates how a request is made according to a users desires, making a request in accordance with our invention with an input screen shown. The content of FIGURE 3 is preformatted 40 except for the user entries which are to be entered in the data input fields 41. In this example the input field 41 is a userid. After a user has entered in field 41 an acceptable input, he would then click on instruction key 42. The instruction key illustrated is submit a request. At this point the Web server captures the information entered by the user, as described in FIGURE 7. It will be appreciated that the Web server captures the information entered by the user, including specialized input, as well as any "hidden" default information, which can include password authorizations, charge account identification, and other information that can be used by the system in responding to the request. Thus the system can assume that the "hidden" password is an authorization to perform some function, such as include information from confidential source, or exit to the Internet. The charge authorization can also be tracked and accumulated by the system as it parses through its functions to charge back chargeable usages. If a request is for an order of an item, the actual item requested can be shipped and billed with this information. Since these functions are "hidden" they do not appear in the FIGURE but included with a request. The return of the request is illustrated in FIGURE 4.

FIGURE 4 is a sample result screen which illustrates how a sample report conforming to the request results are presented to the client after the requested service is provided by the computer system network in accordance with our invention formatted according to the specifications of a DIS capsule which is illustrated by example in FIGURE 9. In this example, the return was a file, whose file name is displayed as P81484 at 43. Informative text accompanying the file is included as illustrated by the example information 44. The screen provides the content of file 43 in the requested form of preformatted text 50 in the form of a display of a text report generated by a DIS capsule stored in the DIS server 17. While we show text as the form the report results, the form of the request can be another form of presentation, as and image, a voice response, or other multimedia presentation. Reports can be returned translated into any desired language based upon the request, as may be provided by DIS capsule calls to a translator. These features are included in the result 50 report.

When the user selected image made by clicking on the image 32 in FIGURE 2, FIGURE 5 appears. FIGURE 5 is a next screen which illustrates how a request is made according to a users desires. A user makes a request, in this instance for sales results within the organization for YTD Catalog Revenue in accordance with our invention by entering text data into the data entry areas 41 and 42 of the formatted screen with information as to type of data selected 40A which will be translated into specific report information created by a DIS capsule.

FIGURE 6 is a sample result screen which illustrates how the request results are presented to the client after the requested service is provided by the computer system network in accordance with our invention formatted according to the specifications of a DIS capsule. In this instance selection of the object 32 links to the the screen of FIGURE 5, which in turn with the DIS capsule created the output shown in FIGURE 6. DIS Capsules will be illustrated by examples in FIGURE 9 and 10. In this example the output of the DIS capsule illustrated in FIGURE 10 is presented on the screen shown by FIGURE 6. The screen comprises a file name identifier, descriptive information 61, and preformatted text 60 which is the display of the named file P555119. This is the display of a graphic report showing what might be deemed (but is not) Confidential information relating to Catalog Revenue for 1995 YTD, with revenue given in \$M, and breakout as to HDW, SFW, PMV, MN and MNT from selected locations in Chicago, New York, Dallas, and Los Angeles. all of which are located on different systems, and which, as illustrated in FIGURE 1, may be on different databases such as DB2, Oracle, and Sybase relational databases. This report was generated by a DIS capsule which is illustrated in FIGURE 9. This example illustrates how multiple actions can be taken on information retrieved. In this example data was translated into image material by calculation and formatting in the form of a graphic pie shaped report. Other image data could also be displayed, as frames of selected images, or a sequence of images in the form of a moving picture display, which can be outputted from a server as will be described in FIGURE 11.

FIGURE 7 illustrates a flowchart showing data flow between a web server and decision support system tool such as IBM's Data Interpretation System (DIS). FIGURE 7 shows the coupling of a Web client 71 (corresponding to Web browser 10 in FIGURE 1) to a

Web server 72 (corresponding to Internet WWW server 11) and the coupling of a request to execute a DIS capsule.

The Web browser 71 can make a request to the Web Server 72 for a report through the use of HTML. The HTML document refers to our control program agent 73, which may be implemented with the C language or other language which can provide run code for the particular Web server which is employed. We illustrate our preferred program according to the description provided in FIGURE 8. The Web Server 72 passes request data to and invokes our control program 73 through the use of the CGI in accordance with our invention. The control program uses ODAS 74 in ODAS server 12 to set DIS capsule parameters and initiates the execution of a DIS capsule located in this embodiment in DIS capsule server 17 according to our preferred examples illustrated in FIGURES 9 and 10.

After a DIS capsule completes execution, the file created by the DIS capsule contains the formatted report results requested by the user. Our control program 73 dynamically creates the HTML statements that present the file to the Web browser 10 screen. Figure 7 shows the coupling within the Web server from ODAS 74 to a distributed DIS LAN 75 with heterogeneous connections to multiple databases DB2, Redbrick, Sybase and Oracle. Other sources of data can be linked to the LAN.

---

### **Preferred Embodiment Interface between Server and DIS**

Our preferred control program agent 73 in FIGURES 1 and 11 is illustrated in detail by way of the flowchart of FIGURE 8. In our preferred embodiment, this program can be written in C or other suitable language but for general appreciation of the details, we will describe the steps in detail. These steps can be implemented by programmers of ordinary skill in the art without undue experimentation after understanding the steps described below. The control program agent 73 is located in a Web server and provides an interface and execution functions. Thus in FIGURE 11 the function is provided between the Web Server 131 (corresponding to Internet WWW server 11 in FIGURE 1) and DIS which is located in a DIS server 133 (corresponding to server 14 in FIGURE 1) and for presentation of results according to the instructions of the Web browser 130 (corresponding to browser 10 in FIGURE 1) according to the request command, which in default is a return to the Web browser home page. This interface utilizes in our preferred embodiment the Web Server CGI and the DIS ODAS.

Before we proceed to the control program 73, it will be noted that in FIGURE 11 the Web Browser 130 will link to a Web Server 131 accessing it on the Internet though a unique ID called the uniform resource locator to access the node which we call the Web server 131. When that access takes place an HTML document is displayed by the Web server 131 to the Web browser 130, as shown in FIGURE 2. Now, the user makes his entries as described with respect to FIGURE 2. Next the HTML document refers to the control program agent 73 and the Web server 131 through the use of the CGI invokes our control program agent 73. The Web server 131 retrieves data entered by

the user from the HTML document and passes that data to our control program agent 73 upon invocation.

The Web Server 131 has a gateway interface that allows the server to invoke a control program agent 73 running on it and to pass input parameters to the control program agent 73 (FIGURE 8) that were returned from the Hypertext document of the Web Browser. It will be appreciated that while we illustrate for our preferred example a single Web Server 131, the Hypertext document locates the particular Web Server that can support the request made by checking the details of the "hidden" defaults and those functions requested. Thus a menu request for a generalized search throughout the Internet may locate the particular service machine having an application processing agent which has the information desired. Once the control program 73 (FIGURE 8) is invoked, the steps programmed for the machine to follow begins with a step 110 illustrated in FIGURE 8. In reviewing this preferred control program agent it should be appreciate that steps 110 and step 111 are steps that are interchangeable in order and which obtain environment variable data from the HTML document return.

Thus step 110 obtains a PATH\_INFO environment variable data. PATH\_INFO contains data from the HTML document that referred the Web Server to our program. Specifically the data contains the name of the DIS capsule to call, the name of the file containing the HTML statements to use when building the HTML document that displays the DIS capsule results to the Web browser, and the type of file that the DIS capsule will create. All off this information is the variable data which is stored in a buffer environment in step 112, and which is used in subsequent steps.

Thus also, the control program proceeds with step 111 which may follow or precede or proceed in parallel with step 110 to obtain the QUERY\_STRING environment variable data. QUERY\_STRING contains data from the HTML document that referred the Web Server to our program. Specifically the data contains values selected by the user and / or default values selected by the HTML document designer. These values are set in the DIS capsule by our control program prior to execution of the DIS capsule. This information is used to set variables in the DIS capsule. All off this information is the variable data which is stored in a buffer environment in step 112, and which is used in subsequent steps.

Within the scope of the discussion of the control program agent illustrated by FIGURE 8 it should be appreciate that the steps 112 through 125 include the utilization of an API set that provides a method of invoking executable programs located in a service machine which we denote as a sub-agent which executes in step 122 object capsules from our sub-agent DIS file server 14. This provides functions such as queue and update functions for databases on multiple platforms and allows the processing of data retried from a database to be performed, including executing calculations, doing formatting, charging of accounts and the storing of results as a file accessible to the control program agent. During processing our control program agent 73 provides setups for API calls which occurs in steps WHAT ARE THESE STEPS. Thus the control program agent will proceed as with an API set with step 113.

With the variable information now stored in a buffer, in step 113 the control program retrieves from a store, all of the DIS capsules that are used and the variable names associated with each DIS capsule and loads into memory associated with the control program the DIS capsule names available and the variable names associated with each DIS capsule.

At that point in step 114 the control program is ready to and does initialize a connection between our control program and the ODAS through the use of an ODAS API. In other environments another API performing similar functions could be used.

At that point, if required for control by the decision support system, and as required by DIS, the control program would log onto the port or desktop for the assigned user. Thus, our control program agent 73 in step 115 logs onto a DIS "desktop", our DIS file server 14.

Once the DIS capsule information is loaded into control program memory, the control program can and does in step 116 retrieve from its memory the DIS capsule variable names associated with the DIS capsule name passed to our control program in the step 110 where PATH\_INFO is provided.

Next, in step 117 the control program creates a data array stored in the control program memory containing the DIS capsule variable names and the values for them that were passed to our control program in the QUERY\_STRING step. These two steps 116 and 117 should be done in order, even though steps 110 and 111 can have an arbitrary order. At this point in step 117 you are matching the DIS capsule variable names with the data that was passed to the control program in the QUERY\_STRING environment variables.

Next, in preparation for a report, in step 118 the program creates a unique filename which may include data originated by the HTML document's variables stored in step 112 (dotted line) to pass to the DIS capsule as a DIS variable for use in naming the report which will be created by the DIS capsule. As a result, the DIS capsule will create that file with the unique file name during its process.

In anticipation of DIS capsule execution, the values of variables used by the DIS capsule are obtained from the data array in the control program memory containing the DIS capsule variable names and the values for them that were passed to our control program in the QUERY\_STRING step. This is done in step 119 using the ODAS API to set the DIS capsule variable values.

At this point the capsule server 17 for the DIS server 133 attached to the Web Server 131 via network 132 will have a DIS capsule services queue. This queue is the queue of jobs being requested of the Dis Capsule Server 17. For the current job request (other like requests being perhaps still in the queue) we use the ODAS API to query the contents of the DIS Capsule Services queue. If the queue size is larger ( $>$ ) than a threshold level, then the process enters a wait state until the queue size is reduced to a tolerable level. The queue test of step 120 is a loop test which returns to test the

queue size until a test answering "is the queue of a size that execution can proceed?" (<t) is answered "YES".

Whenever the queue test is answered YES, at that point the ODAS API is used to submit a DIS capsule for execution in step 121.

After the ODAS API submits a DIS capsule for execution the particular request process being executed by the control program enters a wait state until completion of the DIS capsule execution. For this step of the process the control program uses the ODAS API to wait for completion of the DIS capsule execution performed by the DIS capsule execution 122. During a wait state other requests can be processed by the control program, as requests are fed through the control program as a pipeline, in this WAIT PIPE API step 123, so that the control program continually advances requests through the system.

During the wait state 123 the ODAS API looks for a completion signal. When that is received, the control program then in step 124 reads the file identified by the name passed to the control program in the first PATH\_INFO step that contains the HTML statements which are to be presented with the DIS report results.

While in step 124 the control program reads the file identified, it dynamically creates new HTML statements to display the preformatted text to the Web browser. The new HTML statement include the information retrieved from the file in step 113 so that it can be displayed as a header 44 accompanying the report to be displayed, along with the filename 43.

At this point, in step 125 the control program tests for the kind of report to be created by obtaining information from stored variables and identifies output parameters, such as whether the report is to be a text report, or a graphical report. At this point the control program branches to the sequence applicable to the kind of report to be created. If the output is to be routed to the Web server 10, then the output is routed to the Web server in step 126.

If a text file report is created by the DIS capsule, that determines that a text display is to be reported and the the control program reads the file created by the DIS capsule and dynamically creates HTML statements to display the data lines to the Web browser.

If a graphics file is created by the DIS capsule, that determines that a graphics display is to be reported and the the control program dynamically creates the HTML statement to display the graphics file to the Web browser.

On the other hand, the control program agent allows alternative output direction, and if the output is another type, or an additional output, as for broadcast, it can be routed to another destination. In step 127, we illustrate how using the IBM Digital Server, output can be routed to a requestor selected resulting output selected from a group of possible output units, comprising fax, printer, retail or banking installations, or pro-



vided as a series of full motion videos or still frames which are can be transmitted to display devices, such as a TV set under control of end users with a set-top box cable control. These facilities are provided by providing the output of our control program agent from the web server to the alternative output device 127. In this case, the IBM Digital Server, which with an RS/6000 CPU, Network I/F Bus, DISks, modems, and X.25 Data Switch provides the hardware to route the output to a variety of output devices, to fax, printer, retail, banking, TV or cable customers via the digital server service machine for full motion and still video, supplied with MPEG 2 and MPEG 1 protocol images respectively, to subscribers,

Along the way, the output can be coupled to an auxiliary function, such as back-up or accounting processes 128 which allow for charging for system utilization and service charges for services and items requested. These processes will make use of hidden variables associated with the request, such as charge authorization. One of the hidden variables which may be associated with a request is a credit card number. The credit card number, is preferably encrypted, with a DES or RSA encryption utility, and this along with access authorization variables, will allow access to sensitive databases which reside behind firewalls. If selected data according to the request is permitted to the access authorized user at the location inside or outside the Internet, the data can be included in the results reported by our system to the Web browser.

---

### **Preferred Embodiment of text DIS capsule**

In accordance with our invention, an HTML document, which is running on a web server, refers to the control program agent. The web server then invokes the control program agent. The control program agent has a to command files, which provide the preferred file command objects in the form of DIS capsule objects, or DIS capsules as they are known. The command file contains a list of available DIS capsules. Accordingly, there is not need for the HTML document to know how to get to the command file, as the control program supplies this access. A capsule object, as a DIS capsule, can call other routines which may be written in well known programming languages such as Visual Basic or C. These routines become part of the capsule object by the reference, and these routines perform such functions as account tracking, compression, calculation, handling specific custom outputs such as video, voice. translation, and enable programmability of the capsule objects. The capsule objects also have standard object capability, and we will illustrate these by way of the specific examples described.

It will be seen that the control program 73, described in detail in FIGURE 8 acts in concert with DIS capsule execution. The DIS capsule is an object program with executable additions which we have created to interact with the control program. It should be also understood that the DIS capsule object can perform programmable functions on data which is retrieved from databases. Not only can a DIS capsule get data, it can combine, reformat, and update, the data retrieved. It can act on the data to create new data and basically act as a dedicated processor processing data gathered or created during a Web browser request to output the end result to the user under pro-

programmable parameters determined by the creator of the DIS capsule, as they may be selected, if desired, by the user as part of the request. Thus the user entered inputs as part of his request, either free form or by selection of variables in the menus afforded to the user as illustrated by way of example in FIGURE 5.

DIS capsule objects are like some other objects. For instance in Microsoft's products, an example being the Excel (trademark of Microsoft) spreadsheet, one can click on an object portrayed on the screen and link a succession of objects to perform a specific function, such as take data from a spreadsheet and reformat it into a variety of selectable formats, such as text or graphic illustration. The kind of action to be taken is illustrated by an object on the screen, and linking of routines is done by a succession of clicks on icons representing the object.

In accordance with our preferred embodiment, a DIS capsule is used to invoke system resources. This is done by providing a list of commands, which may be those provided by a DIS processor itself, or written in Visual Basic or C by the programmer. The result is a command file, like an exec or command file in OS/2 or like a \*.BAT file in DOS. These capsules perform the specific functions that are requested by the user from his initiation session. The user further qualifies the execution of the DIS capsule by providing parameters which are used in the invocation.

Now the DIS server 133 supports DIS, the program processor which supports DIS capsules by processing commands contained in the DIS capsule, either directly, in the case of DIS functions, or by to other system or user supplied functions. The user supplied functions comprise mainly those DIS functions which are supplied by DIS and illustrated in the manual "Developing Applications with OpenDIS Access Service, Version 2.0 of the OPEN Access Service." For those not familiar with command files, this manual is fully incorporated herein by this reference as available at the USPTO. An example of a system supplied function would be the base support for SQL queries of a specific database, which are invoked by the DIS capsule program.

In illustrating the specific examples of our invention illustrated in FIGURES 9 and 10, both illustrate linked objects according to a specified flow sequence within a DIS environment. The DIS environment contains numerous functions, including the Internet-work routing functions which the DIS capsules can invoke. Thus, a DIS object which queries a database, as illustrated, invokes the Internetwork routing functions to query databases where they are located on the network. If the preferred example of DIS environment is not supplied, a similar environment with program environment means which supports reaching a destination on the Internet by a link between systems which routes data from one physical unit to another according to the applicable protocol should be supplied. The protocol will employ a URL address for Internet locations.

FIGURE 9 illustrates by way of example a DIS capsule that creates a text report file. Referring to FIGURE 9, it will be seen that the capsule, represented by a series of linked objects, is supported by Internetwork processor support environment means 90. Within this environment an integrated capsule creates a text report file as a result

of the object 95, make text. This object result file is the file 43 according to FIGURE 3 which is displayed at the browser. In the illustrated example, the multiple DIS capsule data retrieval command file 91(a)..91(n) initiates as a first step multiple queries to different databases which are specified by the parameters of the request. In this example, multiple queries are initiated as SQL type search requests as multiple steps 91(a)..91(n) executed by the DIS capsule server 133 with the Database Gateway 134 to select data from DB26000 databases located inside the intranet 140 and on the Internet by Internetwork routing to database gateway 134' and its DB26000 databases by step 91(a). The data is stored in a DIS declared buffer. Similarly, in parallel or successively, additional steps 91(b), 91(c), 91(d), and 91(n) retrieve data and store in their object buffer data retrieved from Sybase, Oracle, Redbrick, and IBM's Data Warehouse databases. Thus object 91(a) will query DB26000 and bring data back to DIS. Object 91(b) will query Oracle and bring data back to DIS. Object 91(c) will query Sybase and bring data back to DIS. Object 91(d) (shown as a dot in FIGURE 9) will query Redbrick and bring data back to DIS, and so on. The nth object 91(n) will query IBM's data warehouse and bring data back to DIS. In a subsequent linked processing step 92 data from the database queries in the first step is joined by joining object command file 92 and stored in a buffer related to this object. Object 92 will joint the data from the n locations searched in step 91. Thereafter, in a subsequent processing step performed by calculation object command file 93 on the joined data in the joined database result buffer of step 92, desired calculations performed in accordance with the parameters indicated by the request are done on the joined data. Thereafter, in accordance with the request parameters text is formatted to space delimited text by the format object command file 94. The results are stored in a buffer associated with format object command file 94. Thereafter, a make text command file 95 causes the formatted text to be created as a text file for the WWW server 131 to be stored in a file which is accessible to and can be retrieved and displayed by the control program agent 73, or directly displayed by the control program agent 73 in the form illustrated in FIGURE 4 at the Web browser 130. It will be noted we have illustrated this process as object capsules in a DIS internetworking environment. These object capsules are a specialized form of a command file, which can encompass additional commands called by an object.

---

### Preferred Embodiment of graphics DIS capsule

FIGURE 10 illustrates by way of example a DIS capsule that creates a graphical report file. For simplicity, data in this FIGURE is also shown in a DIS environment 90. Retrieval object command file 101 illustrates a step of retrieval of data from one or more databases as specified in the parameters of the request, performing these retrieval steps as did retrieval object command files 91(a)..91(n). Thereafter, this data is plotted with the make plot object command file 102, with the results being stored in a buffer. The final step of creating a result- to-be-presented file, in this instance in the form of a bitmap ready for display to a Web browser 130 is created by the make bitmap (BMP) object command file 103. The example of a preferred bitmap object command which would be employed with today's Internet environment is a GIF image. Others can be used as well. Again the results are provided to the Web browser 130, by the action of the program command agent 73 on the Web Server 131, the results

being illustrated by the pie-chart of FIGURE 6 in accordance with the parameters of the request for generating the graphical report illustrated by FIGURE 6.

---

## Alternative Preferred Embodiments

FIGURE 11 illustrates an alternative configuration of the network system as it may be employed for permitting access to information available through homepages and in data warehouses where access to the homepage or database may or may not be restricted by a firewall. In Figure 11, the web browser(s) 130 accesses an associated Web Server 131, 131', 131'' either by a coupling or addressing with a uniform resource locator (URL) the Web Server 131 which may be selected with a Hyperlink. This can be a direct coupling or an indirect coupling, as via a node locatable in a common access medium, such as provided by Internet resources accessible via a web browser, e.g. supporting Web Explorer, or Mosaic, NetScape, node 131 located somewhere on the Internet which utilizes our control program agent 73. Now node 131 which functions as a Web server is coupled via a token-ring network, SNA network, or other suitable network 132 (one of the any which may be used on the Internet as a transmission medium) with the facilities provided within what we will call our intranet, those facilities which are "proprietary" to the owner and which may be protected by firewalls at the intranet boundary 140. Now note that our control program 73 is resident within the Web Server 131 and functions as described in FIGURE 8 to couple to a DIS server 133 located within the intranet 140, which is preferably located behind a firewall as indicated in FIGURE 11. This DIS Server 133 is in turn coupled to our Database gateway 134. This database gateway is configured as illustrated also in FIGURE 1 for gathering information from databases coupled to it and located on servers for DB2, Oracle, Sybase, and Redbrick, as well as one for information warehouse functions. In our preferred embodiments these database units are IBM mainframe systems, as available commercially today, but they could be AS400s, RISC/6000, RISC/6000 SP or other systems supporting the databases.

The DIS Server is a server which supports DIS or similar decision support functions and the functions provided by our DIS capsules illustrated by FIGURE 9 and 10.

Now our Web browsers 130 can not only access information within the intranet, but can reach outside the intranet to gather information located elsewhere via the Internet. We will describe two examples of our preferred couplings to elements on the Internet. One example couples the database gateway 134 to another (a second) database gateway 134' via the Internet and its Internetwork routing (INR) protocol available from IBM as part of its current DIS product which can make use of UALs. The second database gateway 134' is coupled to its own (second) DIS server 133'. At this point the Web browser 130 can access data not only intranet, but also via the Internet to gather data from a database supported by DIS server 133' located outside the intranet. The Database server 134' would be able to gather information from any database coupled to it, as illustrated, assuming access is public or accessible after processing of a hidden variable access authorization.

However, the web browser(s) 130 can also access via Web Server 131 (with our control program 73 illustrated in detail in FIGURE 8) another Web server 131' which implements our control program 73. This Web server, for example, Web server 131' can also be coupled via its own (second) network 132' (which supports functions equivalent to network 132 and as illustrated in FIGURES 1 and 11) to an associated DIS Server 133' as illustrated to perform tasks like those we are describing from a request sent via the second network from its Web server 131'.

However, as another alternative example, Web server 131' with an appropriate API can access a directly coupled database available to the server, such as Microsoft's Access 131a. Thus small databases which have not yet invested in being able to gather information from an intranet resource, can use their own direct resources, and also be interrogated by the Web browser(s) 130, or another web browser 136. Remember that browser's 130 can also communicate with the Web server 131' across the Internet, just as can a Web browser 136 located on the intranet 140 inside of the firewall illustrated by the intranet 140 dashed line shown in FIGURE 11. With a browser 136 in place at the Web Server 131' location, that browser 136 can make requests, if authorized across the intranet to the Web Server 131 which can then utilize the DIS capsules provided by the DIS Server 133.

Physically, the network 132 will have its own access server 135 preferably in the form of a TCP/IP server 135 to make the physical connection across the Internet. We illustrate in FIGURE 11 this other logical layer as located on the network. This TCP/IP server supports the physical connections which are needed by the other logical higher levels of service supported on the network. The use of an InterNetwork Routing Protocol (INR) allows the logical coupling illustrated between a application processing server 134 to an external intranet application processing server 134'. On each network there can be one or more web servers. A Hypertext document request asking for a field to be searched, as by a Hyperlink, could index to a server directly, e.g. a second web server 134'' on the same network which would have its own control program agent function duplicating the control program agent resident in web server 134. Thus at the request homepage a menu which say if "Art&Literature search", when selected in a Hyperlink setting, would index to a particular web server and a particular document within that web server's environment. This web server 134'' besides being linked to its own application processing server 133'' has a direct link, in the environment illustrated, to an MVS CICS, a transaction processing server for handling transaction processing. Such a solution allows CICS transaction processing to utilize the Internet to save transmission costs and still be located beneath a firewall for retention of data integrity. The outputs provided by the web server to the requested destination can be outside of the firewall, and in the form of results illustrated by the possible examples shown in FIGURES 3, 5 and 8.

While we have described our preferred embodiments of our invention, it will be understood that those skilled in the art, both now and in the future, may make various improvements and enhancements which fall within the scope of the claims which follow. These claims should be construed to maintain the proper protection for the invention first disclosed.

PO995020

- 23 -

---

## What is claimed is:

1. A computer network comprising a plurality of servers, each supporting at least one client computer, said network comprising:
  - said client computer (130) for making requests;
  - said server (131) coupled to said client for receiving and fulfilling a request as an agent of said client;
  - a plurality of information access servers (133,133'), for acting as sub-agents for said server during a process of fulfilling requests,
  - said information access servers providing access to capsule objects (FIGS 9 & 10) which perform programmable functions which are executable upon a received command initiated from said server (131),
  - said server (131) including a control program agent (73) for receiving a user request initiated at the client computer (130) for information and for transmitting said request to a sub-agent information access server (133, 133') having capsule objects (DIS capsules) which execute upon control programmable functions requested by said server (131);
  - said sub-agent information access servers (133, 133') being coupled directly and/or via the network to a plurality of database resource gateways (134, 134') for information retrieval from ones of a plurality of database resources (DB2, Oracle, Sybase, Redbrick, Data Warehouse) having data which may fulfill a data need of said request;
  - said sub-agent information access servers (133, 133') executing a capsule object (DIS capsule) to cause any relevant information contained in said plurality of database resources which fulfill a data need of said request to be retrieved and processed by said sub-agent capsule object,
  - said sub-agent after retrieval from the databases and processing of said data storing said retrieved and processed data as results in a file created for return to said control program agent of said server and returning said created file to said server in response to said control program agent transmission,
  - said control program agent of said server upon receipt of said file from said sub-agent causing a report of said results of said sub-agent's processed to a facility determined by said client request.
2. A computer network according to claim 1 wherein
  - said report of said results of said sub-agent's processed to a facility determined by said client request is by default a display of the results of said sub-agent's process to said client for viewing by said user at a web browser.
3. A computer network according to claim 1 wherein
  - said control program agent resides at a server node location accessible by a client computer user with a web browser and comprises a program of instructions executable by a supporting machine environment for performing method steps including receiving a request initiated at said client computer and fulfilling the request by a providing a result obtained from execution of one or more capsule objects executed at sub-agent information access servers (133, 133').

4. A computer network according to claim 1 wherein said control program agent resides at a server node location accessible by a client computer user with a web browser and upon access to the server node of a network, said server displays an HTML document to a user of said client computer; and invoking said control program agent.
5. A computer network according to claim 1 wherein said control program agent resides at a server node location accessible by a client computer user with a web browser and upon access to the server node of a network, said server displays an HTML document to a user of said client computer; and invoking said control program agent; receiving data entered by the user from the HTML document and passing said user entered data to said control program agent upon invocation as input parameters to said control program agent that were returned from said HTML document; said input parameters containing data from the HTML document that referred to said control program agent, including an identification of a command file to call, the name of the file containing the HTML statements to use when building an HTML document that reports results requested by the request, and the type of file that is requested, as well as query data from the HTML document that referred to said control program agent which contains values selected by the user and / or default values selected by a HTML document designer set in a command file program prior to its execution, all of which data is stored in a buffer environment, and accessible by said command file program and / or said control program agent and used to set variables for said command file program.
6. A computer network according to claim 5 wherein an API set is used for invoking executable command files and programs called via said command files.
7. A computer network according to claim 5 wherein said an identification of a command file to call provides the routing to the command file identified.
8. A computer network according to claim 1 wherein said control program agent makes use of an API set that provides for executing object command files for providing programmable functions, including queue and update functions for databases on multiple platforms and allowing the processing of data retrieved from a database to be performed, including executing calculations, doing formatting, charging of accounts and the storing of results as a file accessible to the control program agent.
9. A computer network according to claim 1 wherein said client computer request is initiated with a web browser, said client computer including means for associating said web browser with a homepage including a first control program agent at a homepage node supporting said control program agent coupled to and supporting said homepage and supporting an API to access a database available to said first control program agent node,



said control program agent and API enabling a user of said web browser to gather information from said database available to said first control program agent node and to gather information from an intranet resource and to provide access thereto in response to an interrogation initiated at a remote web browser.

10. A computer network according to claim 1 wherein is included  
means for receiving from a world wide web browser a request to be fulfilled as an agent of the browser client,  
a control program agent for organizing distributed sub-agents as distributed integration solution servers on an intranet network supporting the web server which also has an access agent servers accessible over the Internet.
11. A computer network according to claim 10 wherein is included  
a plurality of distributed integration solution servers for executing selected capsule objects which perform programmable functions upon a received command from said web server control program agent.
12. A computer network according to claim 11 wherein is included  
a database gateway coupled to a plurality of database resources for supplying upon a single request made from a Hypertext document, requested information from multiple data bases located at different types of databases geographically dispersed.
13. A computer network according to claim 12 wherein is included  
command objects for performing calculations, formatting, and other services prior to reporting to the web browser or to other locations, in a selected format a requested result report selected from a set of result reports, including a display report, facsimile report, a printer report, a report to customer installations, and a report to TV video subscribers, with account tracking.
14. A computer network according to claim 1 wherein is included  
a web browser at said client computer for making requests,  
means for associating said web browser with a homepage at said server by a coupling or addressing with a uniform resource locator,  
a control program agent at said server node located somewhere on the Internet supporting said control program agent coupled to and supporting said homepage by a coupling or addressing with a uniform resource locator,  
said control program agent server being coupled via a network with facilities provided within an intranet for private owner facilities and which may be protected by firewalls at the intranet boundary,  
said control program agent being coupled to an information access server functioning as a command file server and said command file server being coupled to a database gateway for gathering information from databases coupled to said database gateway and located on different database servers, said command file server supporting a plurality of command file objects which are programmed to perform web browser service support functions at the request of a user of said web browser to access information within the intranet and to gather information located elsewhere via the Internet as a sub-agent of said control program agent.

15. A computer network according to claim 14,

wherein by submission of a request at a web browser a user can not only access information within an intranet, but can reach outside the intranet to gather information located elsewhere via the Internet.

16. A computer network according to claim 14.

wherein there are on the networking including an intranet and the Internet a plurality of database gateways, and at the command of a command file running within a command file server one database gateway is coupled to another database gateway via the network by an inter-network routing protocol.

17. A computer network according to claim 14,

wherein there are on the networking including an intranet and the Internet a plurality of database gateways, and at the command of a command file running within a command file server one database gateway is coupled to another database gateway via the network by an inter-network routing protocol invoking coupling of database gateways by UALs.

18. A computer network according to claim 16.

wherein a web browser initiated request is distributed via an intranet to the Intranet whereby access of data is obtained not only intranet, but also via the Internet to gather data from a database supported by a command file server located outside the intranet.

19. A computer network according to claim 16,

wherein a web browser initiated request is distributed via an intranet to the Intranet whereby access of data is obtained not only intranet, but also via the Internet to gather data from a database supported by a command file server located outside the intranet and coupled to said command file server with public access or access obtained after processing of variable access authorization data provided through said command file server.

20. A computer network according to claim 1,

a web browser at said client computer for making requests,

means for associating said web browser with a homepage at said server by a coupling or addressing with a uniform resource locator.

a first control program agent at said server located somewhere on the Internet supporting said a control program agent coupled to and supporting said homepage by a coupling or addressing with a uniform resource locator.

said second control program agent node being coupled via a network with facilities provided within an intranet for private owner facilities and which may be protected by firewalls at the intranet boundary,

a second control program agent node located somewhere on the Internet supporting a second control program agent by a coupling or addressing with a uniform resource locator,

said second control program agent node being coupled via a network with facilities provided within an intranet for private owner facilities and which may be protected by firewalls at the intranet boundary.

said first control program agent being coupled to said second control program agent node located somewhere on the Internet supporting said second control program agent and coupled to and supporting a command file server, said command file server being coupled to a database gateway for gathering information from databases coupled to said database gateway and located on different database servers, said command file server supporting a plurality of command file objects which are programmed to perform web browser service support functions at the request of a user of said web browser to access information within the intranet and to gather information located elsewhere via the Internet as a sub-agent of said control program agent.

21. A computer network according to claim 20.

wherein said first control program agent resides on a first web server supporting said web browser and said second control program agent resides on a second web server which is coupled via its own network to an associated command file server to perform tasks requested by said web browser and communicated to said web browser after passing through multiple networks.

22. A computer network according to claim 1, including

a web browser, means for associating said web browser with a homepage including

a first control program agent node supporting a control program agent coupled to and supporting said homepage and supporting an API to access a database available to said first control program agent node,

said control program agent and API enabling a user of said web browser to gather information from said database available to said first control program agent node and to gather information from an intranet resource and to provide access thereto in response to an interrogation initiated at a remote web browser.

23. A computer network according to claim 22.

wherein said remote web browser is also coupled to a second control program agent node located on the Internet, said second control program agent node supporting a second control program agent supporting an API to access a database available to said first control program agent node via said second control program agent

said second control program agent and API enabling a user of said web browser to gather information from a database available to said first control program agent node via said second control program agent node and to gather information from an intranet resource and to provide access thereto in response to an interrogation initiated at said web browser. across the Internet by a coupling or addressing with a uniform resource locator to said second control agent node and from resources available on an intranet coupled to said second control program agent node.

24. A computer network according to claim 22,  
wherein

said second control program agent node is coupled via a network with facilities provided within an intranet for private owner facilities and which may be protected by firewalls at the intranet boundary,

said second control program agent node located somewhere on the Internet supporting said second control program agent by a coupling or addressing with a uniform resource locator,

said first control program agent being coupled to said second control program agent node located somewhere on the Internet supporting said second control program agent and coupled to and supporting a command file server, said command file server being coupled to a database gateway for gathering information from databases coupled to said database gateway and located on different database servers, said command file server supporting a plurality of command file objects which are programmed to perform web browser service support functions at the request of a user of said web browser to access information within the intranet and to gather information located elsewhere via the Internet as a sub-agent of said control program agent.

25. A computer network according to claim 22,

wherein said web browser is at a web server location with said web server providing said control program agent node, and browser requests, if authorized for access across said intranet, accesses a command file agent in a web server on said intranet providing said second command file agent node, which then utilize DIS capsules provided by a DIS Server functioning as a command file server.

26. A computer network according to claim 1, wherein said control program agent has included for controlling receiving and fulfilling requests of a web browser,

a service agent for fulfilling requests of a web browser client coupled to a network; comprising: a control program agent tangibly embodying a program of instructions executable by a supporting machine environment for performing method steps for receiving a request initiated at said web browser client and fulfilling the request by a providing a result, said method steps comprising:

displaying an HTML document to said web browser;

invoking a control program agent;

receiving data entered by the user from the HTML document and passing said user entered data to said control program agent upon invocation as input parameters to said control program agent that were returned from said HTML document;

using an API set for invoking executable command files and programs accessible via a associated command file object.

27. A computer network according to claim 26, wherein said

said input parameters contain data from the HTML document that referred to said control program agent.

28. A computer network according to claim 26, wherein said said step of receiving data entered by the user from the HTML document and passing said user entered data to said control program agent upon invocation as input parameters to said control program agent that were returned from said HTML document includes an

identification of a command file to call, the name of the file containing the HTML statements to use when building an HTML document that reports results requested by the request, and the type of file that is requested, as well as query data from the HTML document that referred to said control program agent.

29. A computer network according to claim 26, wherein said step of passing data from the HTML document that referred to said control program agent, includes passing data which contains values selected by the user and / or default values selected by a HTML document designer set in a command file program prior to its execution, all of which data is stored in a buffer environment, and accessible by said command file program and/ or said control program agent and used to to set variables for said command file program.

30. A computer network according to claim 26, wherein said method steps further comprise:

during processing by said control program agent providing setups for API calls which occur.

31. A computer network according to claim 26, wherein said method steps further comprise:

during processing by said control program agent providing a result according to the request command, which in default is a return to the Web browser home page.

32. A computer network according to claim 1, wherein said a control program agent tangibly embodies a program of instructions executable by a supporting machine environment for performing method steps for receiving a request initiated at said web browser client and fulfilling the request by a providing a result, said method steps comprising:

displaying an HTML document to said web browser;

invoking a control program agent;

receiving data entered by the user from the HTML document and passing said user entered data to said control program agent upon invocation as input parameters to said control program agent that were returned from said HTML document; and storing said input parameters in a buffer;

retrieving with the control program agent from a store command files and the variable names associated with each command files and loading into memory associated with the control program the command file names available and the variable names associated with each command file;

initiating with the control program agent a connection between said control program agent and an API;

once a command file is loaded into memory for said control program agent, the control program agent retrieves from its memory the command file variable names associated with a command file name passed to the control program agent once the control program agent is invoked;

subsequently, creating with said control program agent a a data array stored in the control program memory containing the command file variable names and the values for them that were passed to the control program agent during initial data

retrieval steps in order to match command file variable names with the data that was passed to the control program agent with said environment variables step;

subsequently, in preparation for a report, said program command agent creating creates a unique filename which may include data originated by the HTML document's variables stored in a previous step (step 112) to pass to the command file as a command file variable for use in naming the report which will be created by the command file, which as a result the command file will create that file with the unique file name during its process;

subsequently, in anticipation of command file execution, the values of variables used by the command file are obtained from the data array in the control program memory containing the command file variable names and the values for them that were passed to the control program agent in the step providing parameters for query data from the HTML document that referred to said control program agent which contains values selected by the user and / or default values selected by a HTML document designer set in a command file program prior to its execution, doing this using the API to set the command file variable values;

subsequently, using an API to query and determine the size of a queue of jobs being requested, and comparing whether the size has exceeded a predetermined threshold number of jobs being requested, and if so, then entering a wait state loop to wait and reiteratively comparing whether the size still has exceeded said predetermined threshold number of jobs being requested until the queue size is reduced to below said threshold level, and then at that point using said API submit a command file for execution;

after the API has submitted said command file for execution by a sub agent, said control program agent enters a state of awaiting execution of said command file by said sub-agent;

and wherein, while in step in which the control program agent reads the file identified, it dynamically creates new HTML statements to for a report of results according to a request of the web browser.

33. A computer network according to claim 32 wherein the method steps further comprise: retrieving with said new HTML statement information retrieved from the file so that it can be provided with the results of a command file report.

34. A computer network according to claim 33 wherein the method steps further comprise: testing with said control program agent for the kind of report to be created by obtaining information from stored variables and identifies output parameters, and branching to the sequence applicable to the kind of report to be created.

35. A computer network according to claim 33 wherein the command file agent is a plurality of programmable instructions executable by a supporting machine environment for performing method steps required by a request initiated at a web browser client and fulfilling the request by a providing a result, said method steps comprising: receiving at said command file agent a submit command from a control program agent in preparation for a report and variables associated with a report to pass to the command file as a command file variable for use in naming a report which will be

created by the command file, which as a result the command file will create that file with the unique file name during execution of said command file agent.

processing by said command file agent a series of linked objects according to a specified flow sequence within a distributed data environment specified by said command file, including executing functions specified in said command file agent for including data retrieval and processing, and

in the process creating a report file with said unique file name and storing the result of said processing in said report file having said unique file name during execution of said command file agent.

36. A computer network according to claim 33 having a method which includes method steps of:

invoking internetwork routing functions by said command file sub-agent after authorization data is received during execution of a request submitted by said control program agent.

37. A computer network according to claim 33 having a method which includes method steps of: receiving prior to creating a report file having said unique filename data originated by an HTML document's variables stored by said control program agent.

38. A computer network according to claim 33 having a method which includes method steps of:

invoking internetworking routing functions to query databases where they are located on the network.

39. A computer network according to claim 33 having a method which includes method steps of:

invoking internetworking routing functions to query databases where they are located on the network by reaching a destination on the Internet by a link between systems and routes data retrieved from one physical unit to another according to the applicable protocol.

40. A computer network according to claim 1

wherein said server is a web server for supporting a web browser,  
and said server includes

means for receiving from a world wide web browser a request to be fulfilled as an agent of the browser client,

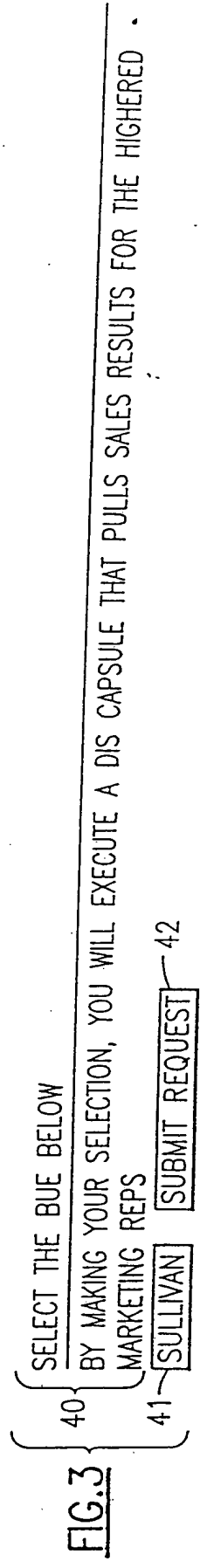
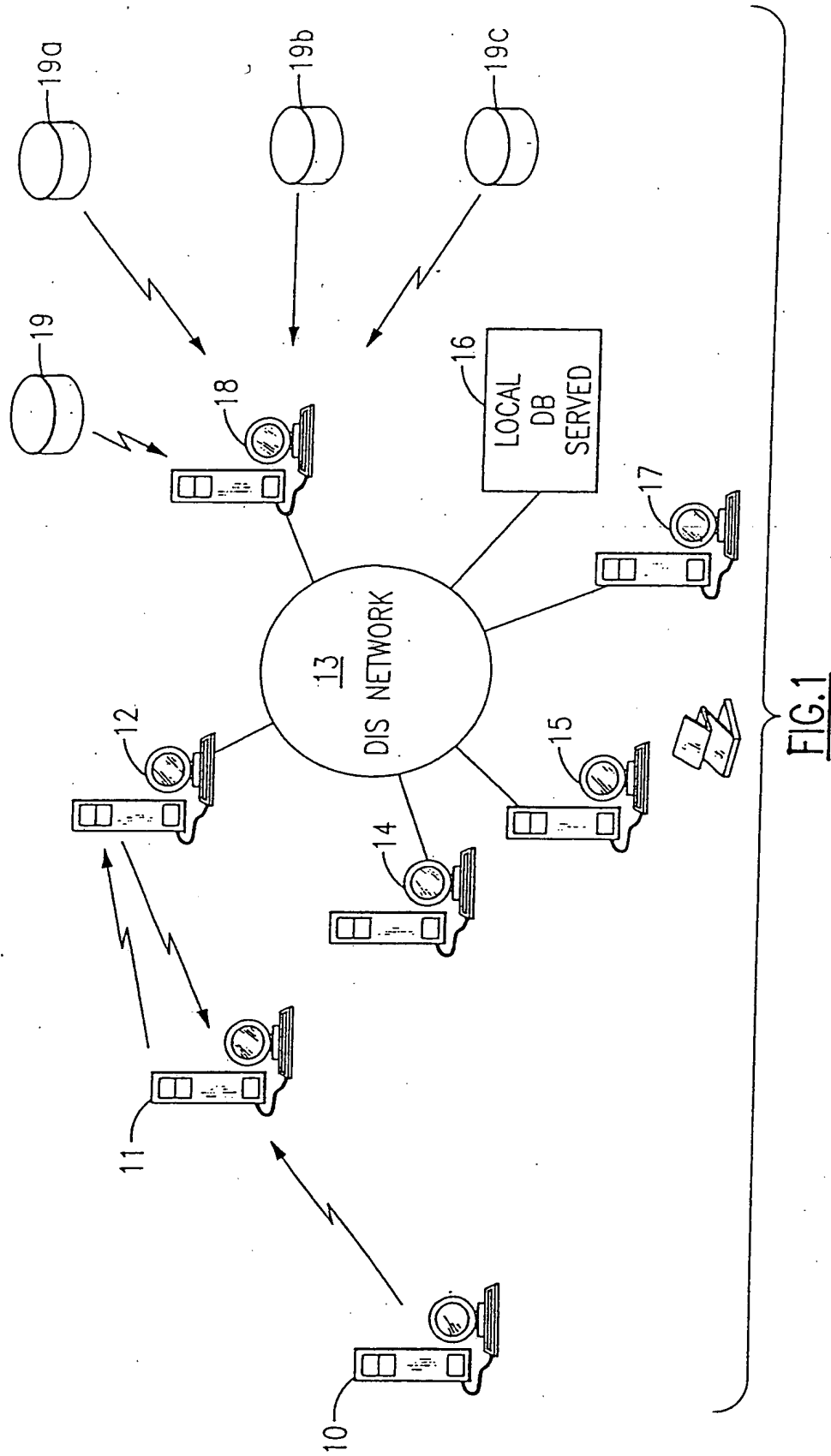
a control program agent for organizing distributed sub-agents as distributed integration solution servers on an intranet network supporting the web server which also has an access agent servers accessible over the Internet.

41. A computer network according to claim 40, further comprising

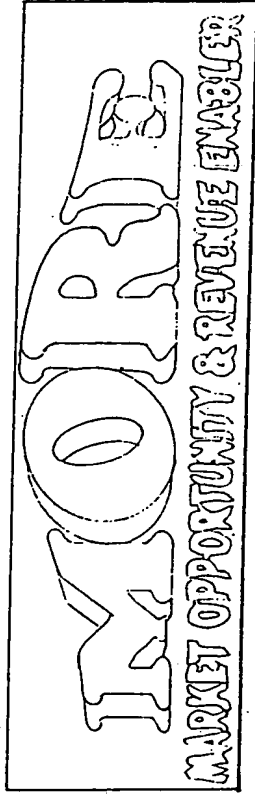
a plurality of distributed integration solution servers for executing selected capsule objects which perform programmable functions upon a received command from said web server control program agent.

42. A computer network according to claim 41, further comprising  
a database gateway coupled to a plurality of database resources for supplying  
upon a single request made from a Hypertext document, requested information from  
multiple data bases located at different types of databases geographically dispersed.
43. A computer network according to claim 41, further comprising  
command objects for performing calculations, formatting, and other services prior  
to reporting to the web browser or to other locations, in a selected format a requested  
result report selected from a set of result reports, including a display report, facsimile  
report, a printer report, a report to customer installations, and a report to TV video  
subscribers, with account tracking.





WELCOME TO THE HOMEPAGE OF...  
29



THE CURRENT DATE IS FRI APR 28 11:05:38 1995

THIS PAGE IS UNDER DEVELOPMENT! (C)COPYRIGHT IBM CORPORATION, 1994, 1995. ALL RIGHTS RESERVED.

"WE'RE COMING TO LIVE FROM THE ON-RAMP TO THE INFORMATION SUPER HIGHWAY." THIS IS THE IBM U.S. HIGHER EDUCATION HOME PAGE ON THE WORLD WIDE WEB.

BY ACCESSING THIS PAGE YOU WILL HAVE THE UNIQUE CAPABILITY OF OBTAINING INTERACTIVE INFORMATION RELATING TO OUR BUSINESS. BY MERELY CLICKING ON THE HIGHLIGHTED TEXT AND FOLLOWING THE ROAD-SIGNS YOU'LL BE WELL ON YOUR WAY TO INFOCENTRIC CITY!!

MAKE YOUR SELECTION BELOW

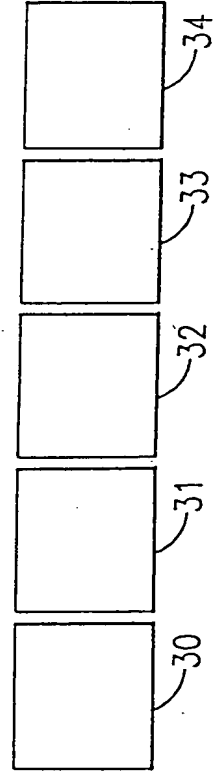


FIG.2

FILE NAME IS P81484. ←43

CATCH IT ON THE NET!

44 YOU JUST RAN A DATA INTERPRETATION SYSTEM CAPSULE!  
THE OUTPUT IS THE TEXT TOOL TRANSFERRED AS A TEXT.  
COPYRIGHT IBM 1995

CUSTOMERS WHO MIGHT WANT TO KNOW ABOUT THE  
DIS WWW GATEWAY---THIS DATA IS FROM THE IBM MARKET PLANNING DATA SYSTEM---  
A DB2 MVS DATABASE

CUSTOMER NAME	SW DESCRIPTION	INST	CITY STATE
AC NIELSEN CO	DIS 1.3.6 DIS ENTERPRISE SE	9501	CHERRY HILL NJ
AC NIELSEN CO	DIS 1.3.6 DIS ENTERPRISE SE	9501	GREEN BAY WI
AC NIELSEN CO	DIS 1.3.6 DIS ENTERPRISE SE	9501	MINNEAPOLIS MN
AC NIELSEN CO	DIS 1.3.6 DIS ENTERPRISE SE	9501	WILTON CT
ADVANTIS	DIS 1.3.6 DIS ENTERPRISE SE	9501	SCHAUMBURG IL
ALTA BATES MEDI	DIS 1.3.6 DIS ENTERPRISE SE	9410	BERKELEY CA
ALTA BATES MEDI	DIS 1.3.6 DIS ENTERPRISE SE	9503	BERKELEY CA
AMERICAN PRESID	DIS 1.3.6 DIS ENTERPRISE SE	9501	OAKLAND CA
ANHEUSER BUSCH	DIS 1.3.6 DIS ENTERPRISE SE	9501	ST LOUIS MO
ANHEUSER BUSCH	DIS 2.0 OPENDIS ACCESS SERV	9501	ST LOUIS MO
50 ASHLAND OIL INC	DIS 1.3.6 DIS ENTERPRISE SE	9502	LEXINGTON KY
ASHLAND OIL INC	DIS 2.0 OPENDIS ACCESS SERV	9502	LEXINGTON KY
BELLSOUTH CELLU	DIS 1.3.6 DIS ENTERPRISE SE	9501	FT LAUDERDALE FL
BELLSOUTH CELLU	DIS 2.0 OPENDIS ACCESS SERV	9501	FT LAUDERDALE FL
BELLSOUTH COMMU	DIS 1.3.6 DIS ENTERPRISE SE	9501	ATLANTA GA
BELLSOUTH COMMU	DIS 1.3.6 DIS ENTERPRISE SE	9501	BIRMINGHAM AL
BELLSOUTH TELEC	DIS 1.3.6 DIS ENTERPRISE SE	9501	ATLANTA GA
BRIO TECHNOLOGY	DIS 1.3.6 DIS ENTERPRISE SE	9501	MOUNTAIN VIEW CA
BRISTOL MYERS S	DIS 1.3.6 DIS ENTERPRISE SE	9410	PLAINSBORO NJ
BRISTOL MYERS S	DIS 2.0 OPENDIS ACCESS SERV	9502	NEW YORK NY
BROADWAY DEPT S	DIS 1.3.6 DIS ENTERPRISE SE	9410	LOS ANGELES CA
BROOLYN UNION	DIS 1.3.6 DIS ENTERPRISE SE	9410	BROOKLYN NY
CHESEBROUGH PON	DIS 1.3.6 DIS ENTERPRISE SE	9501	GREENWICH CT
COLGATE-PALMOLI	DIS 1.3.6 DIS ENTERPRISE SE	9410	IRVINE CA
COLGATE-PALMOLI	DIS 1.3.6 DIS ENTERPRISE SE	9410	MORRISTOWN NJ
COLGATE-PALMOLI	DIS 1.3.6 DIS ENTERPRISE SE	9410	NEW YORK NY

FIG.4

SELECT THE BUE BELOW.

40A

BY MAKING YOUR SELECTION, YOU WILL EXECUTE A DIS CAPSULE THAT PULLS SALES RESULTS FOR YTD CATALOG REVENUE.

41

SULLIVAN

SUBMIT REQUEST

42

FIG.5

FILE NAME IS P555119. ← 59

CATCH IT ON THE NET!

61

YOU JUST RAN A DATA INTERPRETATION SYSTEM CAPSULE! THE OUTPUT IS THE PLOT TOOL TRANSFERRED AS A BITMAP.  
COPYRIGHT IBM 1995

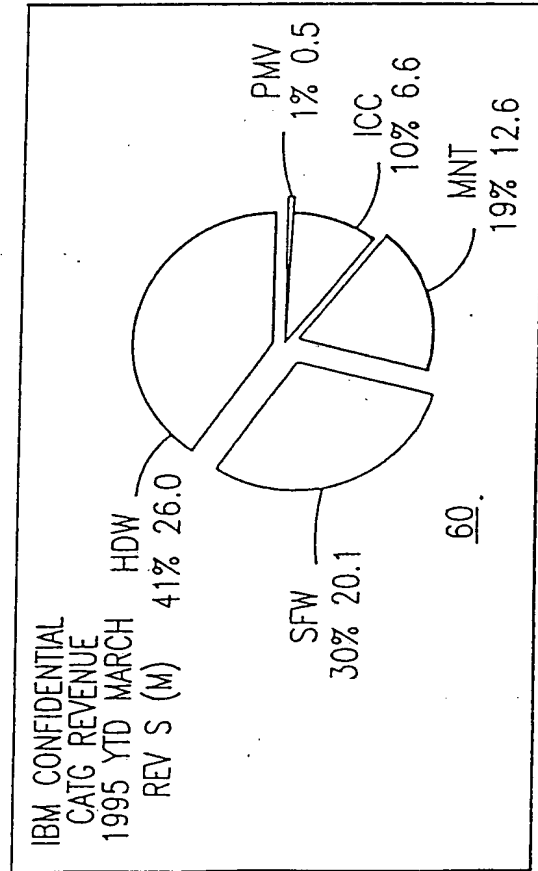


FIG.6

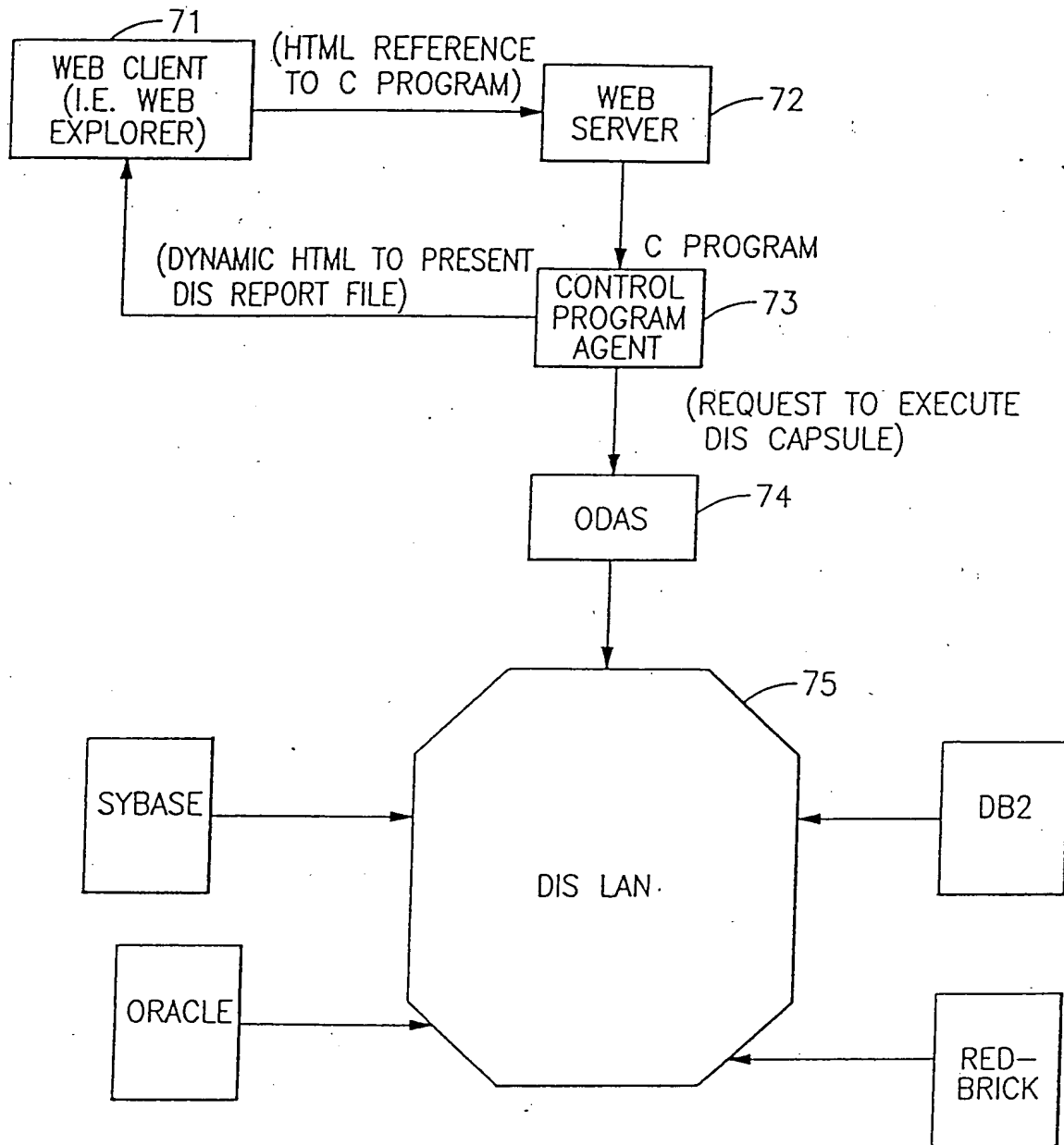
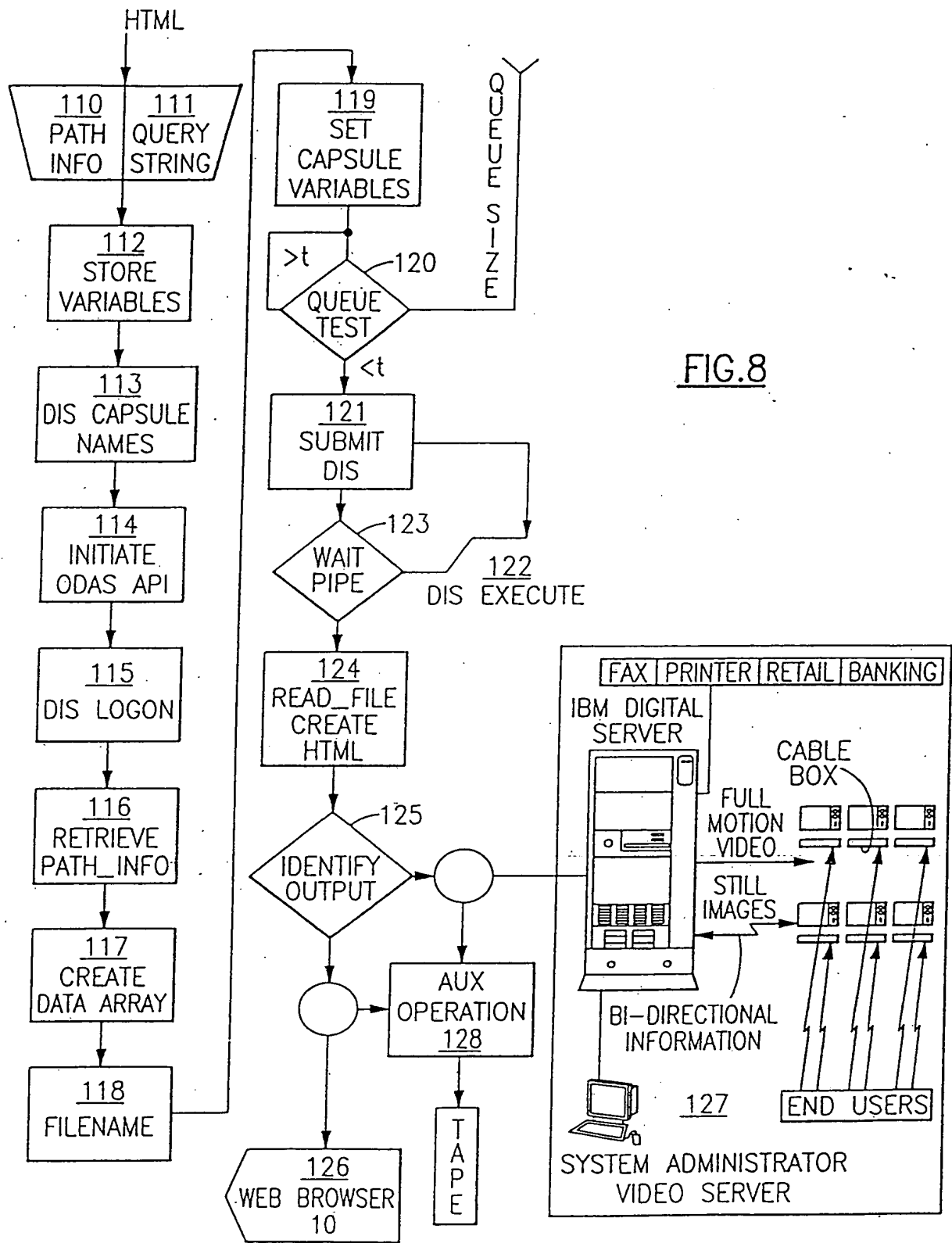


FIG.7



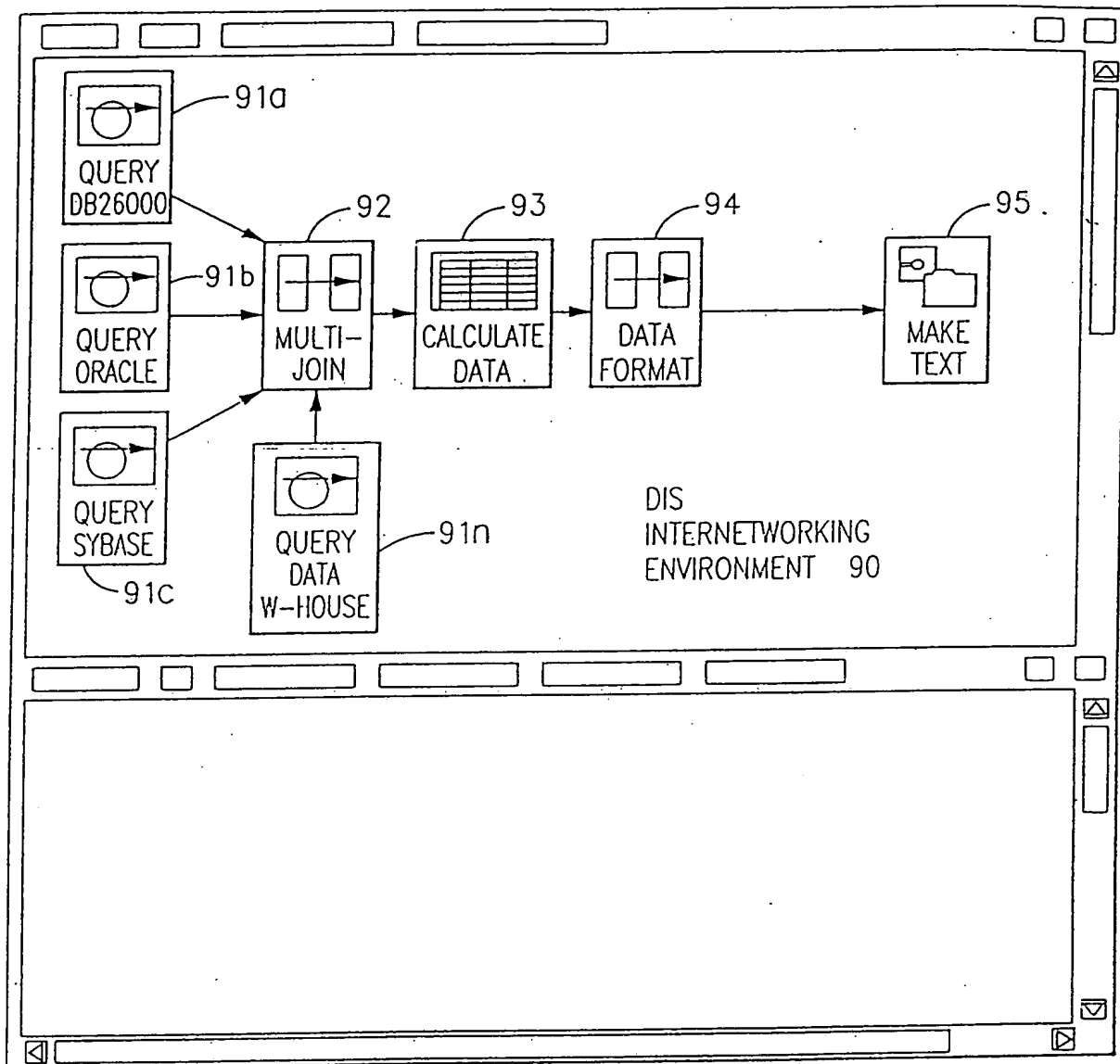


FIG.9

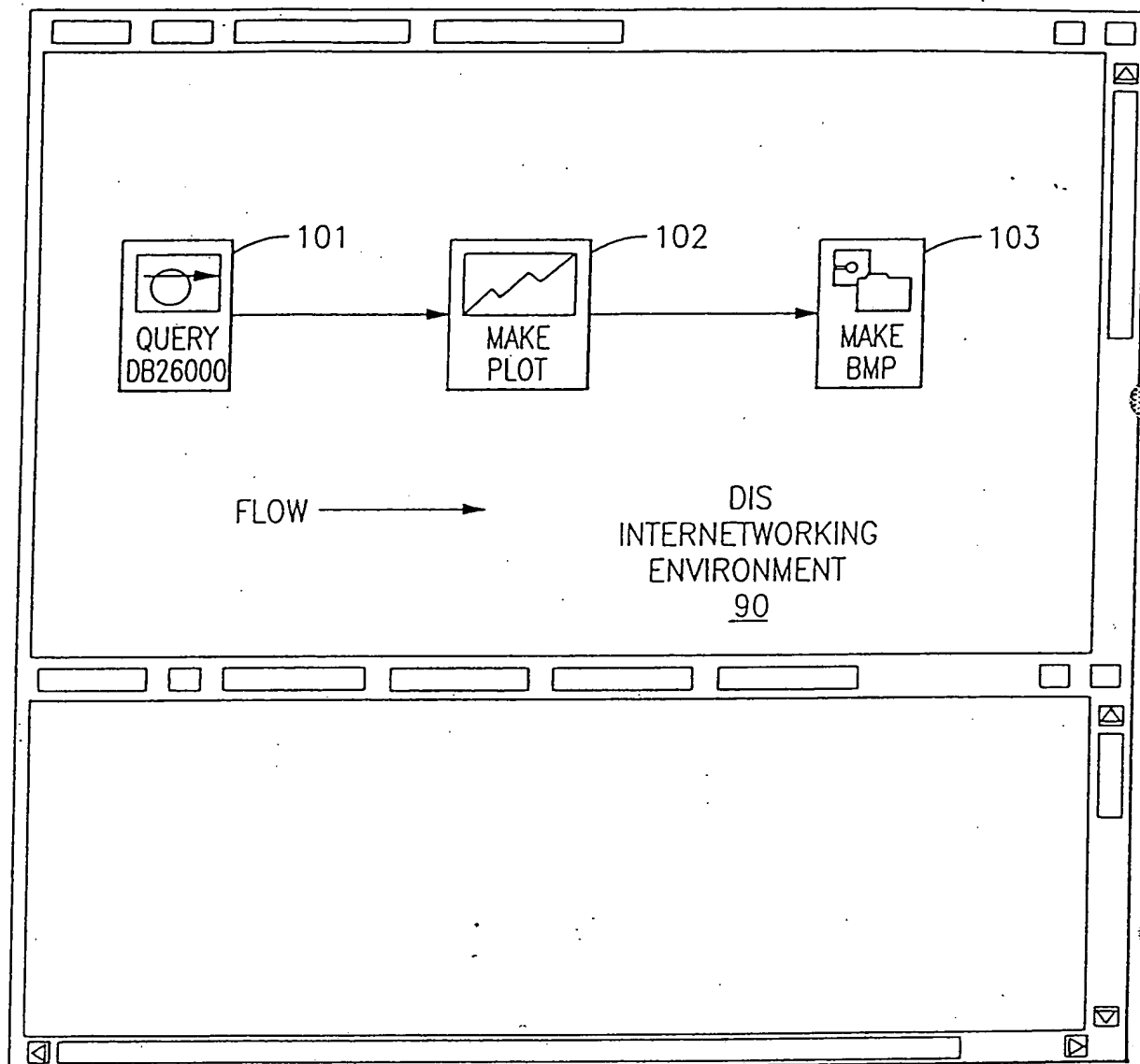
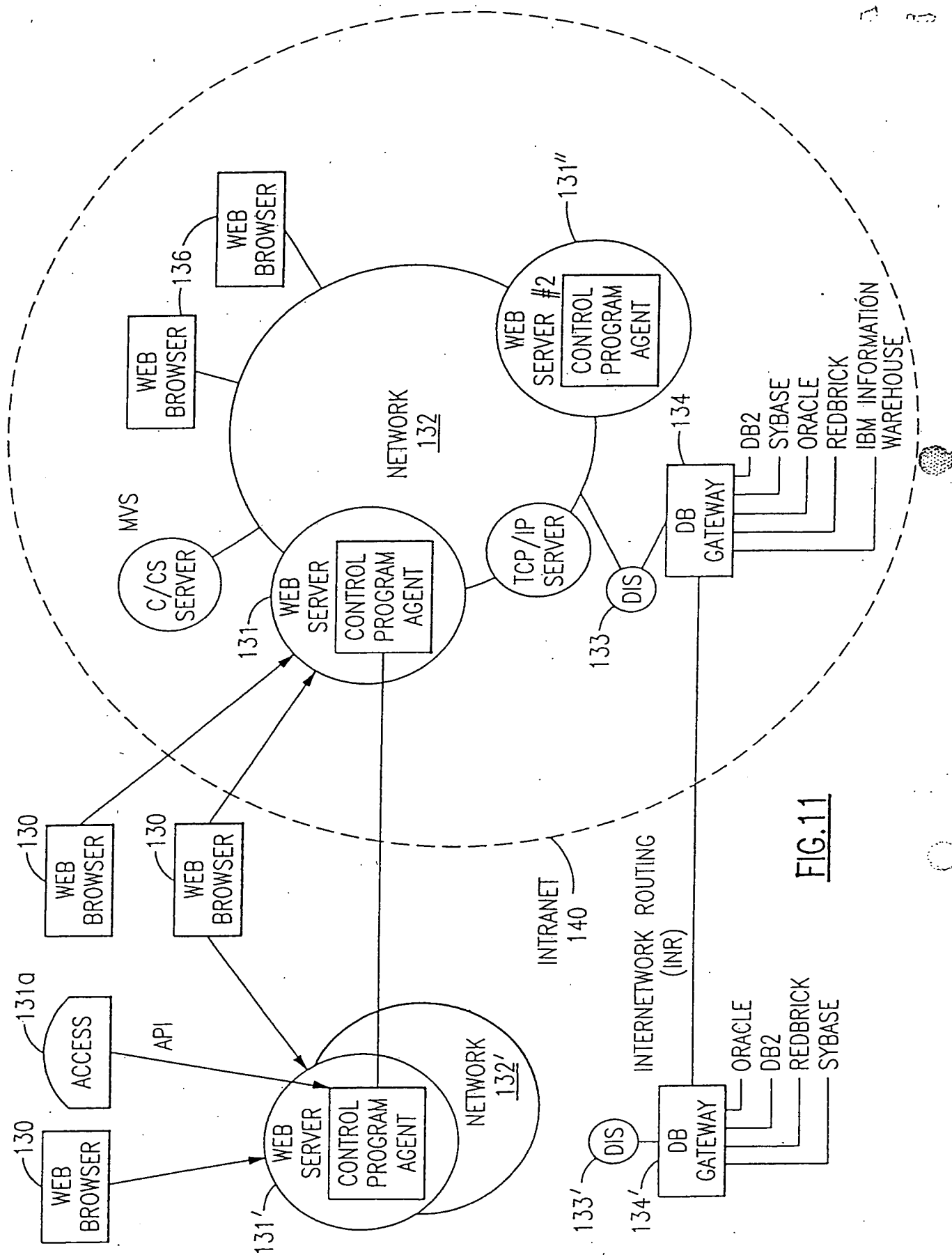


FIG.10





**FIG.11**